Software AG 製品のアップグレード

バージョン 10.3

2018 年 10 月

WEBMETHODS

このマニュアルは、 webMethods Product Suite バージョン 10.3 およびそれ以降のリリースに適用されます。

このマニュアルに含まれる仕様は、変更されることがあります。変更内容については、それ以降のリリースノート、または次のエディションで報告されます。

Copyright © 2011-2018 Software AG, Darmstadt, Germany and/or Software AG USA Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

Software AG およびすべての Software AG 製品の名前は、Software AG, Software AG USA Inc.、またはその子会社やライセンサーの商標または登録商標です。このマニュアルに記載されたその他の企業名および製品名は、それぞれの所有者の商標です。

Software AG とその子会社が所有する商標および特許に関する詳細情報は http://softwareag.com/licenses にあります。

本ソフトウェアはサードパーティ製品の一部を含む場合があります。サードパーティの著作権情報、ライセンス条項、追加の権利や制約については、「License Texts, Copyright Notices and Disclaimers of Third Party Products」 を参照してください。特定のサードパーティのライセンス条項や制約については、「License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products」 の「Legal Notices」の E 項を参照してください。これらのドキュメントは、 http://softwareag.com/licenses またはライセンス製品のルートインストールディレクトリから取得できる製品ドキュメントの一部として参照できます。

Software AG のライセンス許諾書によって特別に明示されていない限り、利用、複製、譲渡、公開および開示は禁止されています。

ドキュメント ID: SAG-UPGR-103-20181015_JA

目次

このマニュアルについて....9
表記規則....9
オンライン情報とサポート....9
データ保護....10

Command Central、Software AG Installer、Software AG Update Manager、および Software AG Migration Utilities を使用したアップグレード....11
Command Central を使用した製品のアップグレード....12
同じインストールディレクトリ、同じホストとポート、稼働中のデータベース....12
新しいインストールディレクトリ、同じホストとポート、クローンデータベース....12
新しいインストールディレクトリと新しいポート、同じホスト、クローンデータベース....12
新しいホスト、稼働中のデータベースまたはクローンデータベース....13
Software AG Installer、Software AG Update Manager、および Software AG Migration Utilities を使用した製品のアップグレード....13

このマニュアルで扱うアップグレード....15
このマニュアルで扱うアップグレード....16
オペレーティングシステム間のアップグレードのサポート....16
リリースの番号付けの例外....16
アップグレードを実行するために必要なマニュアル....16
このマニュアルで扱っていないアップグレード....17

アップグレードを成功させるための重要な要素と要件....19
準備と計画....20
テストとトラブルシューティング....21
一般的なアップグレード手順の要件....22
API Gateway....23
API Portal の要件....23
CentraSite の要件....23
Software AG Designer の要件....23
Integration Server または Microservices Runtime の要件....23
一般的な要件....24
クラスタリングの要件....24
My webMethods Server の要件....24
環境変数....24
My webMethods Server クラスタリングの要件....24
OneData の要件....25
Universal Messaging クラスタリングの要件....26
データベースの要件、推奨事項、および準備....26

新しい製品のインストール....29
Command Central を使用した新しい製品のインストール....30

Software AG Installer を使用した新しい製品のインストール.....30

移行のための新しい製品、古い製品、データベースコンポーネント、および圧縮ファイルの準備.....33
新しい製品への最新の修正のインストール..... 34
Software AG Runtime のシャットダウンおよび自動開始の無効化.....34
移行のための古い環境の準備..... 34
古い製品への修正のインストール.....34
10.1 からのアップグレード : 古い API Gateway の準備.....35
古い API Portal の準備.....35
古い CentraSite の準備..... 37
古い Deployer の準備..... 37
古い Software AG Designer の準備.....37
Integration Server または Microservices Runtime 定義のエクスポート.....37
CloudStreams のサーバ定義とプロジェクトのエクスポート.....37
設定のエクスポート.....38
古い Integration Server または Microservices Runtime の準備.....38
製品の起動と接続..... 38
トリガーを一時停止してキューを空にする..... 38
9.8 からのアップグレード: Command Central を使用してアップグレードする際にトリガーを一時停止してキューを空にする.....38
Command Central を使用しないでアップグレードする際にトリガーを一時停止してキューを空にする..... 39
古い MashZone NextGen の準備..... 39
古い My webMethods Server の準備.....39
古い OneData の準備..... 40
古い Optimize の準備..... 40
古い Zementis Predictive Analytics の準備.....40
古い製品のシャットダウン..... 41
Command Central を使用した古い製品のシャットダウン.....41
古い製品を手動でシャットダウン..... 41
移行するデータベースコンポーネントの準備..... 42
概要.....42
データベースコンポーネントの移行に関する特別な考慮事項..... 42
Trading Networks データベースコンポーネントを移行する前に..... 42
データベース移行スクリプトへの修正のインストール..... 43
移行する古い製品インストールのソースとして圧縮ファイルの作成.....43
Command Central を使用して移行する場合の圧縮ファイルの作成.....43
移行ユーティリティを使用して移行する場合の圧縮ファイルの作成..... 44
My webMethods Server 用の圧縮ファイルの作成.....44
その他すべての製品用の圧縮ファイルの作成..... 44

データベースコンポーネントの移行..... 47
注意.....48
Command Central を使用したデータベースコンポーネントの移行.....48
Database Component Configurator を使用したデータベースコンポーネントの移行.....48
DynamicBusinessOrchestrator データベースコンポーネントの移行.....49

Software AG Infrastructure の移行 51
移行される Software AG Infrastructure の設定、データおよびアセット 52
Software AG Infrastructure の移行 52
Software AG Infrastructure のアップグレードの完了 53

Terracotta の移行 55
Terracotta BigMemory の移行 56
Terracotta DB の移行 56
Terracotta Management Console 設定とデータの移行 56
Ehcache データの移行 56

Universal Messaging の移行 59
移行できる Universal Messaging 設定、データおよびアセット 60
Universal Messaging の移行 60
Universal Messaging のアップグレードの完了 60

My webMethods Server の移行 63
移行できる My webMethods Server 設定、データおよびアセット 64
移行の一部として実行されるその他のアクション 64
埋め込みデータベースのサポート停止 64
My webMethods Server の移行 64
My webMethods Server のアップグレードの完了 65
Java サービスラッパー #include ディレクティブの確認 65
My webMethods Server インスタンスの初期化 65
クローンから稼働中のデータベースに切り替える 66
Business Rules のアップグレードの完了 66

Integration Server、Microservices Runtime、およびホストされている Wm パッケージの移行 67
Integration Server およびMicroservices Runtime 68
パッケージ 68
移行できる設定、データおよびアセット 68
移行の一部として実行されるその他のアクション 69
Integration Server の移行 69
Integration Server のアップグレードの完了 70
Java サービスラッパー #include ディレクティブの確認 70
Integration Server 起動ファイルのカスタマイズ 70
WSDL の更新 70
9.7 または 9.8 からのアップグレード: シリアル webMethods メッセージングトリガーの同期および更新 71
10.1 からのアップグレード: 更新されたパブリッシュ可能なドキュメントタイプの保存および同期 71
Integration Server およびホストされる製品のデータベースコンポーネントへの接続 72
ActiveTransfer のアップグレードの完了 72
CloudStreams のアップグレードの完了 72
TLS 1.1 および 1.2 トラストストアの設定 73
Dynamic Business Orchestrator のアップグレードの完了 74
アダプタのアップグレードの完了 74

webMethods Adapter for JDBC..... 74
webMethods Microsoft Package..... 74
webMethods Adapter for SAP..... 75
eStandards モジュールのアップグレードの完了..... 75
webMethods SWIFTNet Module..... 75

Software AG Designer、ビジネスプロセスランタイムデータ、および Closed Loop Analytics の移行..... 77
Software AG Designer の移行..... 78
移行の前に..... 78
Apama、Application Platform、Business Process および Business Rule プロジェクトの移行..... 78
タスクアプリケーションプロジェクトの移行..... 80
CAF アプリケーションの添付リストコントロールのカスタマイズのやり直し..... 80
CloudStreams のサーバ定義とプロジェクトの移行..... 80
Integration Server または Microservices Runtime 定義の移行..... 81
モバイルプロジェクトの移行..... 81
ビジネスプロセスランタイムデータの移行..... 82
Closed Loop Analytics JMS トピックの移行..... 83

Optimize および Infrastructure Data Collector の移行..... 85
Terracotta Server Array の設定と起動..... 86
他の製品への接続の更新..... 86
Process Engine への接続の更新..... 86
Optimize Support パッケージへの接続の更新..... 86
Infrastructure Data Collector の移行..... 87
移行できる Infrastructure Data Collector 設定、データおよびアセット..... 87
Infrastructure Data Collector の移行..... 87
Infrastructure Data Collector のアップグレードの完了..... 87
Optimize 環境の展開..... 88
古い Optimize CCS 環境定義を再利用しない場合..... 88
古い Optimize CCS 環境定義を再利用する場合..... 88
新しい Optimize 環境の展開..... 89
9.7 からのアップグレード: 4.1 以降の Terracotta Server Array の監視の開始..... 90
webMethods Broker から Universal Messaging への切り替え..... 90

Asset Build Environment、Deployer およびアプリケーションプラットフォームの移行..... 91
Asset Build Environment の移行..... 92
Deployer の移行..... 92
移行される設定、データおよびアセット..... 92
Deployer の移行..... 92
Application Platform の移行..... 92
移行される Application Platform 設定、データ、および、アセット..... 93
Application Platform の移行..... 93
Application Platform のアップグレードの完了..... 93
パッケージのインポートの確認と修正..... 93
Tomcat 設定の更新..... 94

9.8、9.9、9.10 または 9.12 からのアップグレード: WmTomcat Web アプリケーションの移行の完了....94

Apama の移行....97

API Gateway、API Portal、および CentraSite の移行....99
10.1 からのアップグレード: API Gateway の移行....100
移行される API Gateway 設定、データおよびアセット....100
API Gateway の移行....100
API Portal の移行....100
移行できる API Portal アセット....101
API Portal の移行....101
API Portal のアップグレードの完了....102
CentraSite の移行....102
移行される CentraSite 設定....102
CentraSite の移行....103
CentraSite のアップグレードの完了....103

MashZone NextGen の移行....105
10.1、または 10.2 からのアップグレード: MashZone NextGen の移行....106
移行できる MashZone NextGen 設定、データおよびアセット....106
MashZone NextGen の移行....106
MashZone NextGen のアップグレードの完了....107
9.10、または 9.12 からのアップグレード: MashZone NextGen の移行....108
MashZone NextGen リポジトリの設定....108
埋め込みの Derby データベースを使用する場合のリポジトリの設定....108
その他のデータベースを使用する場合のリポジトリの設定....109
9.10 からのアップグレード: 古いリポジトリから新しいリポジトリへのデータの追加....110
MashZone NextGen の設定の更新と拡張機能のコピー....111
外部設定ディレクトリを使用していた場合の更新方法....111
外部設定ディレクトリを使用していなかった場合の更新方法....111
9.10 からのアップグレード: フィードの移行とアップグレードコマンドの実行....111

9.7、9.8、または 9.9 からのアップグレード: Mobile Designer の移行....113

OneData の移行....115
移行できる OneData 設定、データおよびアセット....116
OneData の移行....116
OneData のアップグレードの完了....116

Zementis Predictive Analytics の移行....117
移行される Zementis Predictive Analytics アセット....118
Zementis Predictive Analytics の移行....118

イベントインフラストラクチャの移行....119
Digital Event Services の移行....120
移行される Digital Event Services 設定、データおよびアセット....120

Digital Event Services の移行.....120
Digital Event Services のアップグレードの完了.....120
Event Routing の移行..... 120
移行される Event Routing 設定、データおよびアセット.....120
Event Routing の移行.....121
Event Routing のアップグレードの完了.....121
イベント永続化サービスの移行..... 121

移行ユーティリティを使用した製品の移行.....123
移行ユーティリティを実行する前に..... 124
Software AG インフラクトラクチャ移行ユーティリティの実行.....124
Universal Messaging 移行ユーティリティの実行.....125
カスタムの移行の実行..... 125
インポートした設定を使用した移行.....125
プロンプトを表示して、カスタムのインポートされた設定を使用して移行する.....126
プロンプトを表示せずに、カスタムのインポートされた設定を使用して移行する (サイレント).....126
プロンプトを表示せずに、デフォルトのインポートされた設定を使用して移行する (サイレント)...126
My webMethods Server 移行ユーティリティの実行.....127
カスタムの移行の実行..... 127
インポートした設定を使用した移行.....127
プロンプトを表示して、カスタムのインポートされた設定を使用して移行する.....128
プロンプトを表示せずに、デフォルトのインポートされた設定を使用して移行する (サイレント)...128
プロンプトを表示せずに、カスタムのインポートされた設定を使用して移行する (サイレント).....128
Integration Server 移行ユーティリティの実行.....129
カスタムの移行の実行..... 129
インポートした設定を使用した移行.....131
プロンプトを表示して、カスタムまたはデフォルトのインポートされた設定を使用して移行する.... 131
プロンプトを表示せず、カスタムまたはデフォルトのインポートされた設定を使用して移行する (サイレント).....132
Infrastructure Data Collector 移行ユーティリティの実行..... 132
カスタムの移行の実行..... 132
インポートした設定を使用した移行.....133
プロンプトを表示して、カスタムまたはデフォルトのインポートされた設定を使用して移行する.... 133
プロンプトを表示せずに、カスタムまたはデフォルトのインポートされた設定を使用して移行する. 133
Application Platform 移行ユーティリティの実行..... 134
MashZone NextGen 移行ユーティリティの実行..... 134
カスタムの移行の実行..... 134
インポートした設定を使用した移行.....134
プロンプトを表示して、カスタムのインポートされた設定を使用して移行する.....135
プロンプトを表示せず、カスタムまたはデフォルトのインポートされた設定を使用して移行する (サイレント).....135
OneData 移行ユーティリティの実行..... 135
Digital Event Services 移行ユーティリティの実行.....136
Event Routing 移行ユーティリティの実行.....136

全製品の最終的なアップグレードタスクの完了..... 137

このマニュアルについて

このマニュアルでは、Software AG 製品のアップグレード手順について説明します。

表記規則

規則	説明
太字	画面上の要素を表します。
縮小フォント	*folder.subfolder.service* 形式のサービス名や場所、API、Java クラス、メソッド、プロパティを表します。
斜体	次の情報を表します。 独自の状況または環境に固有の値を指定する必要がある変数。 本文で最初に出現する新しい用語。 他のドキュメントソースへの参照。
モノスペースフォント	次の情報を表します。 入力する必要があるテキストまたはシステムで表示されるメッセージ。 プログラムコード。
{ }	選択肢のセットを表します。ここから 1 つ選択する必要があります。中カッコの内側にある情報のみを入力します。{ } 記号は入力しません。
|	構文行で相互排他的な 2 つの選択肢を区切ります。いずれかの選択肢を入力します。| 記号は入力しません。
[ ]	1 つ以上のオプションを表します。大カッコの内側にある情報のみを入力します。[ ] 記号は入力しません。
...	同じ種類の情報を複数回入力できることを示します。情報だけを入力してください。実際のコードに繰り返し記号 (...) を入力しないでください。

オンライン情報とサポート

Software AG マニュアルの Web サイト

マニュアルは、Software AG マニュアルの Web サイト (「http://documentation.softwareag.com」) で入手できます。このサイトには、Software AG の製品サポートサイト Empower のクレデンシャルが必要です。Empower のクレデンシャルがない場合は、TECHcommunity Web サイトを使用する必要があります。

Software AG の Empower 製品サポート Web サイト

Empower のアカウントをまだお持ちでない場合は、氏名、会社名、および会社の電子メールアドレスを「empower@softwareag.com」 まで送信し、アカウントをリクエストしてください。

アカウントを取得したら、Empower (「https://empower.softwareag.com/」) の [eService] セクションからオンラインでサポートインシデントを開くことができます。

製品情報は、Software AG の Empower 製品サポート Web サイト (「https://empower.softwareag.com」) で入手できます。

機能および拡張機能に関するリクエストの送信、製品の可用性に関する情報の取得、製品のダウンロードを実行するには、「Products」 に移動します。

修正に関する情報を取得し、早期警告、技術論文、Knowledge Base の記事を読むには、「Knowledge Center」 に移動します。

ご不明な点がありましたら、「https://empower.softwareag.com/public_directory.asp」 の [Global Support Contact Directory] でお住まいの国のローカルまたはフリーダイヤル番号を検索し、お電話でお問い合わせください。

Software AG TECHcommunity

マニュアルおよびその他の技術情報は、Software AG TECHcommunity Web サイト (「http://techcommunity.softwareag.com」) で入手できます。以下の操作を実行できます。

- TECHcommunity クレデンシャルを持っている場合は、製品マニュアルにアクセスできます。TECHcommunity クレデンシャルがない場合は、登録し、関心事の領域として [マニュアル] を指定する必要があります。
- 記事、コードサンプル、デモ、チュートリアルにアクセスする
- Software AG の専門家によって承認されたオンライン掲示板フォーラムを使用して、質問したり、ベストプラクティスを話し合ったり、他の顧客が Software AG のテクノロジをどのように使用しているかを学んだりすることが可能です。
- オープンスタンダードや Web テクノロジを取り扱う外部 Web サイトにリンクできます。

データ保護

Software AG 製品は、EU 一般データ保護規則 (GDPR) に従って個人データの処理に関する機能を提供します。該当する場合、それぞれの管理マニュアルに適切な手順が記載されています。

1 Command Central、Software AG Installer、Software AG Update Manager、および Software AG Migration Utilities を使用したアップグレード

■ Command Central を使用した製品のアップグレード 12

■ Software AG Installer、Software AG Update Manager、および Software AG Migration Utilities を使用した製品のアップグレード 13

Command Central を使用した製品のアップグレード

Command Central を使用して単一環境で製品をアップグレードする場合は、このマニュアルの手順に従ってください。

Command Central を使用して、複数の似たような環境の製品アップグレードを完全に自動化するには、*Software AG Command Central Help* のコンポジットテンプレートに関する手順を参照してください。テンプレートの移行セクションのあとにコマンドを追加することで、ホスト名やポートなどのエンドポイントを再設定することができます。スクリプト化できない場合は、このマニュアル内の手動タスクを実行する必要があります。

Command Central を使用したアップグレードや移行をサポートする製品を確認するには、『*Software AG Command Central Feature Support Matrix*』を参照してください。

メモ: Platform Manager をアップグレードする必要はありません。

Command Central は、以下のアップグレードシナリオをサポートします。

同じインストールディレクトリ、同じホストとポート、稼働中のデータベース

この場合、Command Central は、古い製品のインストールディレクトリ名を変更し、新しい製品のインストールで古いディレクトリ名を使用します。

利点: 簡単。再インストールと同じように、すべてのパス、ホスト、およびポートが同一のものを使用できる。カスタムコンポジットテンプレートを使用して、完全自動化が可能。

注意点: リスクを伴う。ロールバックが簡単にできない。大規模なダウンタイムが発生する。稼働中のデータベースのバックアップが必須である (クローンデータベースの使用はサポートされません)。古いインストールが存続するため、倍のディスク容量が必要。稼働環境を移行する前に、広範囲の自動化テストが必須。

新しいインストールディレクトリ、同じホストとポート、クローンデータベース

利点: 新しいインストールは古いインストールとほぼ同一。1 番目のシナリオよりも安全 (ロールバックが比較的簡単)。新しいインストールのシャットダウン後に古いインストールの起動が可能 (またはその逆も可能)。カスタムコンポジットテンプレートを使用して、完全自動化が可能。あるいは、Command Central Web ユーザインタフェースやコマンドを組み合わせて段階的に実行することもできる。

注意点: 古いインストールが存続するため、倍のディスク容量が必要。新しいインストールディレクトリをポイントするために、サードパーティ製ツールを調整する必要がある。

新しいインストールディレクトリと新しいポート、同じホスト、クローンデータベース

利点: 1 番目のシナリオよりも安全 (ロールバックが比較的簡単)。1 番目や 2 番目のシナリオよりもダウンタイムが少ないので、古いインストールを稼働しながら段階的に移行が可能。カスタムコンポジットテンプレートを使用して、完全自動化が可能。

注意点: 1 番目や 2 番目のシナリオよりも移行準備が複雑。古いインストールが存続するため、倍のディスク容量が必要。エンドポイントを変更して新しいポートを反映させる必要がある。新しいインストールに対してサードパーティ製ツールの調整が必須。

新しいホスト、稼働中のデータベースまたはクローンデータベース

このシナリオは通常、ハードウェアやオペレーティングシステムのアップグレードで必要となります。同じインストールディレクトリとポートを使用することで、変更を最小限に抑えられます。また、新旧の環境で IP とホスト間のマッピングに DNS を使用することで、外部クライアントへの影響が少なくなります。

利点: 最も安全なシナリオで、ダウンタイムもあまりかからない。プライベートクラウドまたはパブリッククラウドで環境が実行される場合に最も簡単なシナリオであり、一時的に容量を増やすことで、新しい環境をホストしてから古い環境で使用されたリソースを解放できる。段階的に移行が可能。新しいインストールに対してサードパーティ製ツールの調整が必須。カスタムコンポジットテンプレートを使用して、完全自動化が可能。あるいは、Command Central Web ユーザインタフェースやコマンドを組み合わせて段階的に実行することもできる。

注意点: オペレーティングシステム間の移行が可能な製品もあるが、その他の製品では、このような移行によって予期しない結果が起こる場合がある。特に古いマシンが Windows で、新しいマシンが Windows 以外のマシン (またはその逆) である場合は注意が必要。

Software AG Installer、Software AG Update Manager、および Software AG Migration Utilities を使用した製品のアップグレード

Software AG Installer を使用した新しい製品のインストール、Software AG Update Manager を使用した修正のインストール、あるいは Software AG 移行ユーティリティを使用した製品設定の移行でアップグレードを行う場合は、このマニュアルの手順に従ってください。

2 このマニュアルで扱うアップグレード

- このマニュアルで扱うアップグレード 16
- オペレーティングシステム間のアップグレードのサポート 16
- リリースの番号付けの例外 16
- アップグレードを実行するために必要なマニュアル 16
- このマニュアルで扱っていないアップグレード 17

このマニュアルで扱うアップグレード

このマニュアルでは、Presto 9.7、9.8、9.9、9.10、9.12、10.1、および 10.2 から 10.3 へのアップグレード方法について説明します。

- Apama、MashZone NextGen、Universal Messaging、および webMethods 製品。
- Terracotta および Zementis Predictive Analytics (webMethods 製品と併用する場合)。

このマニュアルの手順を使用して、10.3 インストールから別の 10.3 インストールにデータを移行することもできます (データセンターへの移動など)。

オペレーティングシステム間のアップグレードのサポート

Integration Server、Microservices Runtime、OneData、および Universal Messaging は同じオペレーティングシステムを搭載した複数のマシン間での移行、および、異なるオペレーティングシステムを搭載したマシン間での移行をサポートします。

他の製品では、Windows システムから別の Windows システムへの移行、および、UNIX システムから別の UNIX システムへの移行をサポートします (両システムが同じベンダーの JVM を使用している場合)。これらの製品では、あるオペレーティングシステムから異なるオペレーティングシステムへの移行はサポートされません (たとえば、Windows システムから UNIX システムへの移行、またはこの逆方向の移行もサポートされません)。

リリースの番号付けの例外

以下の製品は、一般的なリリースの番号付けに従っていません。

- webMethods 9.9、9.10、および 9.12 リリースでは、ActiveTransfer は 9.8 リリースのままです。
- webMethods の 9.7、9.8、9.9、9.10、9.12、10.1、10.2、および 10.3 リリースでは、Terracotta のリリース番号はそれぞれ 4.1.4、4.3、4.3.1、4.3.2、4.3.3、4.3.4、4.3.5、および 4.3.6 です。

アップグレードを実行するために必要なマニュアル

アップグレードを実行するには、下記のマニュアルが常に必要になります。

- 新機能が記載されている古いリリースの次のリリースから新しいリリースまでのリリースノート。
- 新しいリリースの製品の Readme。これらには、製品動作の変更、サービス、API、その他の項目の廃止、削除、追加、新しい製品リリースに関する重要な情報、既知の問題と解決された問題が記載されています。
- 新しい製品にインストールする修正の Readme。これらには、解決された問題と動作の変更が記載されています。

- *System Requirements for Software AG Products*
- *Supported Upgrade Paths for Software AG Products*
- *Software AG Command Central Help* または *Using Software AG Installer*
- *Installing Software AG Products*
- *Using Software AG Update Manager* (最新リリース)

このアップグレードを実行するには、アップグレードする製品に応じて下記のマニュアルが必要になります。新リリースのマニュアルが必要です。

- *webMethods API Portal Administrator's Guide* および *webMethods API Portal Customization Guide*
- *webMethods Application Platform User's Guide*
- *webMethods BPM Task Development Help*
- CentraSite のマニュアル
- *webMethods Deployer User's Guide*
- *webMethods Integration Server Administrator's Guide*、*webMethods Service Development Help*、*webMethods Integration Server Clustering Guide*
- *MashZone NextGen Administration Guide* および *MashZone NextGen User and Developer Guide*
- *Administering My webMethods Server*
- *Administering webMethods OneData*
- *Administering webMethods Optimize*、*Configuring BAM*
- *Using Terracotta with webMethods Products* および Terracotta BigMemory Max のマニュアル
- Universal Messaging

このマニュアルで扱っていないアップグレード

以下の表はこのマニュアルで扱っていないアップグレードを一覧し、正しい情報ソースを提供しています。

アップグレード	情報
9.7 以前のリリースから	『*Supported Upgrade Paths* 』を参照してください。
9.9 以前の Apama リリース	9.9 以前の Apama リリースノートを参照してください。
ApplinX	ApplinX のマニュアルを参照してください。

アップグレード	情報
webMethods Broker から Universal Messaging	webMethods Broker は廃止されました。webMethods Broker 9.6 から Universal Messaging 10.3 への移行は可能です。この移行を計画するには、『*Migrating from webMethods Broker to webMethods Universal Messaging*』を参照してください。移行中は、webMethods Broker 9.6 と Universal Messaging 10.3 を、同じ Integration Server 10.3 と共に使用し続けることができます。
Command Central	『*Software AG Command Central Help*』を参照してください。
Presto および MashZone NextGen	Presto 3.8 (webMethods リリース 9.7)、Presto 3.9 (webMethods リリース 9.8)、または MashZone NextGen 9.9 から MashZone NextGen 10.3 への直接のパスはありません。代わりに、9.10 のアップグレードガイドにある手順を使用して MashZone NextGen 9.10 へアップグレードし、その後、このガイドにある手順を使用して、9.10 から MashZone NextGen 10.3 へアップグレードする必要があります。
Terracotta (webMethods 製品と併用しない場合)	Terracotta BigMemory Max のマニュアルを参照してください。
Terracotta BigMemory Max から Terracotta DB	Terracotta DB 10.3 は、クラスタトポロジやデータ記憶領域の形式、その他の機能の操作の点で、Terracotta BigMemory Max 4.x および 3.7.6 とは大きく異なります。Terracotta BigMemory Max サーバのデータと設定を Terracotta DB サーバへ移行することはできません。
Zementis Predictive Analytics (webMethods 製品および 10.1 より前のリリースと併用しない場合)	アップグレードを行うリリースの『Zementis installation guide』を参照してください。

3 アップグレードを成功させるための重要な要素と要件

- 準備と計画 20
- テストとトラブルシューティング 21
- 一般的なアップグレード手順の要件 22
- API Gateway 23
- API Portal の要件 23
- CentraSite の要件 23
- Software AG Designer の要件 23
- Integration Server または Microservices Runtime の要件 23
- My webMethods Server の要件 24
- OneData の要件 25
- Universal Messaging クラスタリングの要件 26
- データベースの要件、推奨事項、および準備 26

準備と計画

Software AG では、このマニュアルに記載された手順を実行する前に、計画および準備を行うことをお勧めします。計画および準備には以下のタスクが含まれる可能性があります。

- このマニュアルの全体を読んで、実行する必要があるすべてのタスクについての理解を深めてください。
- 製品のリリースノートをお読みください。リリースノートには、新機能に関する情報が記載されています。ご使用の古いリリースの次のリリースから新しいリリースまでのすべてのリリースに関する情報を参照してください。
- Software AG Infrastructure の Readme を含む、新しいリリースの製品 Readme をお読みください。すべての Readme は、Software AG Documentation websiteから入手できます。製品の Readme には次のような情報が含まれます。
  - 製品の重要な情報、既知の問題と解決された問題。
  - 製品動作、サービス、パラメータ、プロパティ、および API への変更。このような変更には、追加、変更、廃止、削除が含まれます。これらの変更に対応するために、移行後に製品ファイルまたはアセットの変更が必要になることがあるので、この情報は特に重要です。

  ご使用の古いリリースの次のリリースから新しいリリースまでの製品の Readme を参照してください。たとえば、9.8 から 10.3 にアップグレードする場合は、リリース 9.9 から 10.3 までの情報を参照してください。
- Empower Product Support Website の 「 Knowledge Center」 に移動し、技術記事を読みます (例えば KB 1788395)。「upgrade」のようなキーワードを使用して、関連する技術記事と、古いリリース、新しいリリース、その間のリリースを検索します。
- アップグレード手法とカットオーバー計画を定義します。大規模なアップグレードを行う場合は、段階的なアップグレードが必要なことがあります。
- 組織内のすべての影響を受けるグループを対象とする通常のプロジェクト管理要件に関する運用ガイドラインを設定します。
- ソースコントロールのアプローチを定義し、ターゲットの環境へのデータの移行をサポートするリリース管理ツールおよび手順を設定します。
- Software AG 製品と通信し、エンドポイント URL、ホスト名、IP アドレスの更新を必要とするクライアントなどの Software AG 環境の外部要因について考慮します。
- アップグレードの一部として、古い環境のベースラインを取得し、そのベースラインを使用してアップグレード手順を実行します。Software AG では、ベースラインから新しいライブラリ環境に移行する間はコードフリーズにすることをお勧めします。
- 環境設定を忘れると、アップグレードに関する多くの問題が発生する原因になります。これらを完全に計画、文書化、およびテストします。
- 広範囲のシステムテストが必要になる複雑なプロジェクトの場合、広範囲のコードフリーズが不可能な場合があります。この場合は、古い環境で発生するコードの変更を追跡するための変更管理戦略を定義

します。ベースラインのテストの完了後、これにより、コードの変更をベースラインにマージして、新しいベースラインを再テストすることが可能になります。

計画アプローチの作成方法に関するサポートが必要な場合は、Software AG Global Consulting Services までご連絡ください。「Software AG TECHcommunity」 にアクセスして、アップグレード Wiki でヒントや他の情報を見つけることもできます ([Wiki] > [webMethods] > [Upgrade])。

テストとトラブルシューティング

テストには以下のタスクが含まれる可能性があります。

- Software AG では、実稼動環境に移行する前に、制御されたテスト環境でアップグレードを行い、システムが適切に動作するかどうかを検証することを強くお勧めします。
- アップグレードをサポートするための新しい環境をプロビジョニングします。テストを実施する環境では、実稼動環境を複製する必要があります。
- ソース環境からテストデータを取得して、テスト結果のベースラインを確立します。
- 環境設定を忘れると、アップグレードに関する多くの問題が発生する原因になります。これらを完全に計画、文書化、およびテストします。
- ユニットテスト、機能テスト、非機能回帰テストのテストツールを識別します。
- 許容条件とサインオフ条件を定義します。
- アップグレードの回帰テストのコアセットを自動化します。Software AG では、最も多くの変更を加えられた製品領域について最も重点的にテストを行うことをお勧めします。Readme とリリースノートには、すべての製品変更情報が記載されています (詳細については、20 ページの「準備と計画」を参照してください)。
- 新しいマシンにアップグレードする場合は、システムテストを開始する前にバックエンドシステムへの接続とファイアウォールが開かれていることを確認してください。
- 問題管理システムを使用して欠陥と問題を追跡します。
- テスト中の古い環境のコードフリーズが不可能である場合は、古い環境からのコード変更をベースラインにマージして新しいベースラインを再テストします。

テストアプローチの作成方法に関するサポートが必要な場合は、Software AG Global Consulting Services までご連絡ください。

アップグレード時にエラーが発生した場合、以下のトラブルシューティング方法を試してください。

- ログファイルを確認します。

  以下の表には、アップグレード情報とその場所を記録しているログのタイプが一覧されています。

ログのタイプ	場所
インストールとアンインストール	*new_Software AG_directory* /install/logs ディレクトリ

ログのタイプ	場所
製品	製品ファイル構造内の logs ディレクトリ、または *new_Software AG_directory* /profiles/*product* /logs ディレクトリ
データの移行	*new_Software AG_directory* /install/logs ディレクトリ、およびアップグレード手順で示された製品ディレクトリ
データベースの移行	*new_Software AG_directory* /common/db/logs ディレクトリ

- Empower Product Support Websiteの「Knowledge Center」 に移動し、技術記事を読みます (例えば KB 1788395)。「upgrade」のようなキーワードを使用して、関連する技術記事と、古いリリース、新しいリリース、その間のリリースを検索します。
- 「Software AG TECHcommunity」 に移動します。
  - アップグレードディスカッションフォーラムに参加します ([Forums] > [webMethods] > [Upgrade])。
  - アップグレード Wiki でヒントや他の情報を見つけたり投稿したりすることもできます ([Wiki] > [webMethods] > [Upgrade])。

上記の方法で問題を解決できない場合、Software AG Global Support までご連絡ください。

一般的なアップグレード手順の要件

重要: これらの要件に合致しない場合は、予期しない結果になり、インストールやデータの破損が起きることがあります。

- アップグレードマニュアルのタスクは、示されている順序で正確に実行してください。ご使用の製品には、共有されているインフラストラクチャやイベント駆動型アーキテクチャなどの多くの相互依存性があるため、タスクの順序は重要です。
- 新しい製品をインストールした後に、最新の修正をインストールします。ただし、アップグレード手順が完了するには数日または数週間かかることがあります。手順の実行中に新しい修正を継続的にチェックして定期的にインストールします。
- このマニュアルで指示があるまで、新しい製品を起動しないでください。起動すると、データベースコンポーネントが壊れる可能性があります。
- 指定のアップグレードパスのみで必要なタスクであると明記されていない限り、このマニュアルで説明するタスクはすべてのアップグレードパスについて実行してください。
- このマニュアルでは、古い環境で使用していた RDBMS を新しい環境でも使用することを想定しています。新しい環境で異なる RDBMS を使用する場合は、製品をアップグレードする前に、特別な要件を満たす必要があります。詳細については、Software AG Global Consulting Services に連絡してください。

API Gateway

メモ: Event Data Store の名称は、10.2 で Internal Data Store へ変更されました。

API Gateway はさまざまなオペレーティングシステムをサポートしています。ただし、API Gateway は Windows および Red Hat Enterprise Linux オペレーティングシステムのみをサポートする Internal Data Store に依存しています。このため、2 つの新しい API Gateway をインストールすることも考えられます。その場合、1 つは移行された API Gateway アセットをホストし、もう 1 つは移行された Event Data Store データと共に Internal Data Store をホストします。

API Portal の要件

API Portal をアップグレードするには、以下の機能特権が必要です。

- ライセンス、ユーザおよびドキュメントの管理
- ユーザ管理設定
- データベース管理

API Portal 管理者に問い合わせて、ユーザ管理コンポーネント (http://*host* :*port* /umc) にログオンし、ユーザ管理ページに移動し、自分のユーザ名をクリックします。次に、自分を API Administrator グループに追加するか、または [Functional Privileges] タブに移動して前述の特権を自分に割り当てます。

CentraSite の要件

CentraSite Application Server Tier および CentraSite Registry Repository を異なるマシンにインストールするか、または同じディレクトリに異なるタイミングでインストールする場合は、まず Registry Repository をインストールする必要があります。

Software AG Designer の要件

Software AG Designer プロジェクトに関連する製品をアップグレードしている場合は、移行するプロジェクト用の十分な容量がターゲットマシンにあることを確認してください (77 ページの「Software AG Designer、ビジネスプロセスランタイムデータ、および Closed Loop Analytics の移行」参照)。

Integration Server または Microservices Runtime の要件

メモ: Microservices Runtime の手順は Integration Server の場合と同じです。

一般的な要件

移行するカスタムパッケージ用の十分なスペースがターゲットマシンにあることを確認してください。カスタムパッケージは、Software AG Designer でユーザが作成したパッケージおよび Software AG Designer からユーザが生成したビジネスプロセスパッケージを含みます。

クラスタリングの要件

Integration Server では、クラスタは、同じ ISInternal データベースコンポーネントをポイントし、Terracotta Server Array の同じキャッシュマネージャに接続する複数の Integration Server インスタンスとして定義されます。クラスタは、異なるマシン上の Integration Server インストールか、同じ Integration Server インストール内のサーバインスタンス、またはその両方を含むことができます。

このセクションは、アップグレード手順の残りの部分と共に使用することを意図しています。使用する製品を対象に、マニュアル内のすべてのタスクを記載された順番で実行する必要があり、追加のタスクまたは例外が以下に記載されています。

- 新しい Integration Server をインストールする場合は、古いクラスタに一致するクラスタを作成します。一連の新しい Integration Server は並行してインストールできます。
- 新しい Integration Server に修正をインストールする場合、並行してインストールすることができます。
- 古い製品をシャットダウンする際に、すべてのクラスタノードをシャットダウンします。

*webMethods Integration Server Clustering Guide*の手順に従って、クラスタを設定します。

My webMethods Server の要件

環境変数

古いリリースで、環境変数 WM_HOME を使用して、インストールのホストマシンのグローバルパスにデフォルトの My webMethods Server インストールを指定した可能性があります。古いリリースと同じマシンに新しいリリースをインストールする場合、この変数の存在により問題が発生することがあります。したがって、グローバルパスからこの環境変数を削除する必要があります。

新しい My webMethods Server のインストール後に、この環境変数を再定義しないでください。

My webMethods Server クラスタリングの要件

My webMethods Server では、クラスタは、同じ MywebMethodsServer データベースコンポーネントをポイントする複数の My webMethods Server インスタンスとして定義されます。クラスタは、異なるマシン上の My webMethods Server インストールや My webMethods Server インストール内のサーバインスタンスのいずれか、または両方を含むことができます。

このセクションは、アップグレード手順の残りの部分と共に使用することを意図しています。使用する製品を対象に、マニュアル内のすべてのタスクを記載された順番で実行する必要があり、追加のタスクまたは例外が以下に記載されています。

- Universal Messaging をインストールしてから My webMethods Server をインストールします。

  メモ: 10.0 から、My webMethods Server クラスタは、JMS イベントを交換するために、または、JMS プロバイダとして、サーバデータベースを使用していません。代わりに、Universal Messaging がクラスタノードの同期のために必須になりました。

- 新しい My webMethods Server をインストールする場合は、古いクラスタに一致するクラスタを作成します。一連の新しい My webMethods Server は並行してインストールできます。
- 新しい My webMethods Server に修正をインストールする場合、並行してインストールすることができます。
- このマニュアルの手順どおりに Software AG Infrastructure をアップグレードする必要はありません。
- 古い製品をシャットダウンする際に、すべてのクラスタノードをシャットダウンします。
- 古い My webMethods Server インストールから新しい My webMethods Server インストールにサーバインスタンスを移行する場合は、並行して移行できます。
- サーバインスタンスを初期化する場合は、並行して初期化できます。

*Administering My webMethods Server*の手順に従って、クラスタを設定します。

OneData の要件

1. *old_Software AG_directory* /profiles/ODE/configuration/com.softwareag.platform.config.propsloader ディレクトリに移動します。com.softwareag.catalina.connector.http.pid-*port* .properties ファイルおよび com.softwareag.catalina.connector.https.pid-*port* .properties ファイルを開き、ポート番号を書きとめます。
2. 下記のディレクトリに移動して、指定のファイルを開きます。
   - 9.7 または 9.8 からのアップグレード: *old_Software AG_directory* /profiles/ODE/configuration/tomcat/conf ディレクトリに移動し、context.xml ファイルを開きます。
   - 9.9 以降からのアップグレード: *old_Software AG_directory* /profiles/ODE/configuration/com.softwareag.platform.config.propsloader ディレクトリに移動し、com.softwareag.catalina.resource.pid-*repository_name schema* .properties ファイルを開きます。*repository_name* は、インストール中に指定したリポジトリ名で、*schema* は、md、wa、または ra です。

   メタデータ、作業領域、リリース領域スキーマの名前の先頭は、データベース接続プレフィックスです。たとえば、メタデータスキーマの名前 jdbc/dev2md では、接続プレフィックスは dev2 です。スキーマに指定される maxIdle、maxActive、maxWait などのデータベースパラメータもあります。

3. *old_Software AG_directory* /profiles/ODE/bin/onedata/config ディレクトリに移動します。repository.xml ファイルを開き、リポジトリ ID (RepositoryID 要素) と リポジトリ名 (Name 要素) に注目します。

Universal Messaging クラスタリングの要件

Universal Messaging では、クラスタは、クラスタリング用に設定された同一または異なるマシン上の Universal Messaging インストールの、複数のサーバインスタンスとして定義されます。

このセクションは、アップグレード手順の残りの部分と共に使用することを意図しています。使用する製品を対象に、マニュアル内のすべてのタスクを記載された順番で実行する必要があり、追加のタスクまたは例外が以下に記載されています。

- 新しい Universal Messaging サーバインスタンスをインストールする場合は、古いクラスタに一致するクラスタを作成します。新しい Universal Messaging サーバインスタンスは、並行してインストールできます。
- 新しい Universal Messaging サーバインスタンスに最新の修正をインストールする場合は、並行してインストールすることができます。
- 古い製品をシャットダウンする際に、すべてのクラスタノードをシャットダウンします。
- 古い Universal Messaging インストールから新しい Universal Messaging インストールにサーバインスタンスを移行する場合は、並行して移行できます。
- 新しいクラスタノードが古いクラスタノードと同じホストマシンにある場合は、新しいクラスタノードをすべて起動します。

  新しいクラスタノードが古いクラスタノードとは異なるマシンにある場合は、以下の手順に従います。

  1. 1 つのクラスタノードの *new_Software AG_directory* /UniversalMessaging/server/*instance_name* /bin ディレクトリに移動し、remote_realms_bootstrap.conf という名前のファイルを作成します。*instance_name* .address=*new_host_name_or_IP* 形式 (たとえば、umserver1.address=myUM1.com や umserver2.address=192.168.0.1) で、クラスタの一部である各ノードのファイルに、クロスホストのマッピングプロパティを追加します。ファイルを、ほかのすべてのノードの同じ場所にコピーします。
  2. すべてのクラスタノードを起動します。各インストールの remote_realms_bootstrap.conf ファイルは、remote_realms_bootstrap_old.conf という名前に変更されます。ソース Universal Messaging サーバインスタンスが特定のネットワークインタフェース (絶対 IP アドレス) にバインドされている場合、移行されたインスタンスアドレスのインタフェースは 0.0.0.0 (ポート上の既知の全インタフェース) にバインドします。
  3. Enterprise Manager を開いて、クラスタが正しく実行されていることを確認します。

Universal Messaging マニュアルの手順に従って、クラスタを設定します。

データベースの要件、推奨事項、および準備

- 使用している RDBMS を『*System Requirements for Software AG Products*』と照らし合わせて確認します。使用している RDBMS のバージョンが新しい製品でサポートされていない場合は、サポートされている RDBMS バージョンにアップグレードしてください。
- 一部の製品 (Integration Server、Microservices Runtime、Optimize、Process Engine、Trading Networks など) には、データをデータベースコンポーネントからアーカイブまたは消去するための機能があります。データをアーカイブまたは消去すれば、後でデータベースコンポーネントを移行する際にかかる時間を短縮できます。手順については、製品マニュアルを参照してください。
- 製品データベースのバックアップを作成します。バックアップを作成する前にデータベースに接続しているすべての ActiveTransfer、Integration Server、Microservices Runtime、My webMethods Server、OneData、および Optimize インスタンスをシャットダウンします。My webMethods Server をアップグレードする場合は、データベースをバックアップするときに、同時に My webMethods Server インストールディレクトリをバックアップします。問題が発生した場合は、両方のタイプのバックアップからデータを復元する必要があります。
- Software AG は、アップグレードをテストする際にクローン化されたデータベースの使用をお勧めします。新しい環境で使用するデータベースコンポーネントのみをクローン化できます。クローンの操作は、通常、製品データベースコンポーネントを含むスキーマ (Oracle の場合) またはデータベース (SQL Server および DB2 の場合) のレベルか、すべてのデータベースコンポーネントを所有するデータベースユーザのレベルで行われます。データのクローンは、通常、データベースにバンドルされているエクスポートとインポートのツールを使用して実行します。クローンの手順については、データベースベンダーのマニュアルを参照してください。

  Software AG では、次の手順をお勧めします。

  - 『*Installing Software AG Products*』では、製品データベースコンポーネントの動作に必要な基本的な権限と特権について説明します。データベースのクローンを作成する前に、クローン化されたデータベースで作業するデータベースユーザに対して、稼働中のデータベースで作業するデータベースユーザに付与されているものと同じ基本的な権限および特権を付与します。
  - 一部のケースでは、1 人のデータベースユーザが 2 人目のユーザに許可を付与します (たとえば、プロセス監査データベースユーザが、プロセス監査データをアーカイブするために、2 人目のユーザに許可を付与します)。データベースのクローンを作成する前に、新しいスキーマまたはデータベースで 2 人目のユーザを作成します。
  - 別のデータベースユーザ (Oracle の場合) またはデータベース (SQL Server または DB2 の場合) を使用して、クローン化されたデータベースをホストします。
- 新しいリリースでは、新しいテーブル、列、キー、インデックスなど、データベースコンポーネントの変更が必要になる場合があります。新しい製品リリースと互換性を持つように既存のデータベーススキーマを更新するデータベース移行スクリプトを実行します。スクリプトは、既存のデータベースコンポーネントを変更するか、新しい構造を持つ並行データベースコンポーネントを作成して、テーブル、列、キー、インデックスを INSERT、SELECT、RENAME および DROP します。これらの変更で、データベースのサイズが増える場合があります。
- データベースの移行は、通常アップグレードの重要なパスであり、データの量によっては数時間かかることがあります。

4 新しい製品のインストール

- Command Central を使用した新しい製品のインストール 30
- Software AG Installer を使用した新しい製品のインストール 30

Command Central を使用した新しい製品のインストール

Command Central を使用して新しい製品や修正をインストールするには、『*Software AG Command Central Help*』の手順に従ってください。

Software AG Installer を使用した新しい製品のインストール

Software AG Installer を使用して新しい製品をインストールするには、『*Using Software AG Installer*』および『*Installing Software AG Products*』を参照し、以下の手順に従ってください。

Software AG Installer を実行している場合:

1. インストールディレクトリには、新しい Software AG インストールディレクトリを指定します。
2. 製品ツリーで、インストールする新しい製品と [Database Configuration] を選択します。
3. 製品パネル、または、プロンプトへの応答で、必要なデータを指定します。

メモ: 古い製品と同じマシンに新しい製品をインストールする場合、古い製品が実行中であっても、通常は、古い製品によって使用されているポートを新しい製品に割り当てることができます。同じポートを割り当てると、新しいリリースを使い始める際に、アセットとクライアントでポート値を編集する必要がありません。

以下の表は、指定された製品について、Installer パネル、または、プロンプトへの応答で指定するデータを表しています。

製品	ガイドライン
ActiveTransfer Server	データベース接続を新しい ActiveTransfer Server で使用する ActiveTransfer データベースコンポーネントにポイントします。
Apama	新しい Apama を古い Apama と同じマシンにインストールする場合は、新しい作業ディレクトリを指定します。Software AG では、作業ディレクトリ名には、デフォルトで表示されるようにリリース番号を含めることをお勧めします。
Integration Server または Microservices Runtime	新しい Integration Server または Microservices Runtime で使用する ISInternal データベースコンポーネントにデータベース接続をポイントします。

製品	ガイドライン
My webMethods Server	[**アップグレード用サイドバイサイドのインストール**] を選択します。インスタンスは作成されません。
OneData	あらかじめ記録しておいた、古いインストールのポート、リポジトリ ID、リポジトリ名、データベース接続プリフィックス、パラメータの値を入力します。新しい OneData で使用する OneData データベースコンポーネントをデータベース接続がポイントしていることを確認します。 **メモ:** ポート値を再利用すると、古い OneData の Web サービスクライアントを新しい OneData にアクセスするために URL を変更する必要がなくなります。
Trading Networks	新しい Trading Networks で使用する TradingNetworks データベースコンポーネントで、データベース接続をポイントします。
Universal Messaging	[**アップグレード用サイドバイサイドのインストール**] を選択します。インスタンスは作成されません。

5 移行のための新しい製品、古い製品、データベースコンポーネント、および圧縮ファイルの準備

- 新しい製品への最新の修正のインストール 34
- Software AG Runtime のシャットダウンおよび自動開始の無効化 34
- 移行のための古い環境の準備 34
- 移行するデータベースコンポーネントの準備 42
- 移行する古い製品インストールのソースとして圧縮ファイルの作成 43

新しい製品への最新の修正のインストール

『*Installing Software AG Products*』の手順には、新しい製品の最新の修正のインストールが含まれます。

- Common Platform および Security Infrastructure という名前の Infrastructure コンポーネントを除き、すべての新しい製品に最新の製品修正をインストールします。データベーススクリプトには最新の修正をインストールしないでください。この手順の後半で製品修正を適用します。
- 一部の製品では、移行に関連する修正は、製品の修正とは分けられています。Common Platform および Security Infrastructure を含むすべての新しい製品に最新の移行の修正をインストールします。通常、これらの製品の修正名には、それぞれ文字 OSGI と SIN が含まれます。
- 最新の移行フレームワークの修正をインストールします。移行フレームワークの修正名は、MIG_*release* _MigrationFramework_Fix*number* の規則に従っています。[Common Library] の下に表示されます。

重要: アップグレードが完了までに数日または数週間要する場合、その期間、新しい修正があるかどうか定期的に確認し、修正をインストールしてください。

Software AG Runtime のシャットダウンおよび自動開始の無効化

Software AG Runtime はインストール後、自動的に開始されます。Windows サービスまたは UNIX デーモンを停止して Software AG Runtime をシャットダウンします。*new_Software AG_directory* /profiles/CTP/bin ディレクトリの shutdown.{bat|sh} スクリプトを使用できます。

Windows システムに新しい製品をインストールして、製品を Windows サービスとしてインストールした場合、サービスの起動モードはデフォルトで [自動] になります。この手順で起動が指示される前に、新しい製品が誤って起動することを防ぐため、サービスを [手動] に設定します。同様の理由で、UNIX システムにインストールして、デーモンを自動的に起動するスクリプトがある場合、そのスクリプトを無効にします。

重要: この時点では新しい製品を起動しないでください。このマニュアルで明確な指示があるまで、新しい製品を起動してはいけません。起動すると、データベースコンポーネントが壊れる可能性があります。

移行のための古い環境の準備

古い製品への修正のインストール

古い ActiveTransfer、API Gateway、API Portal、CentraSite、Infrastructure Data Collector、My webMethods Server、Optimize、および Process Engine に最新の製品修正をインストールします。古い製品に修正をインストールしたら、その古い製品を起動します。

10.1 からのアップグレード：古い API Gateway の準備

この節では、以下の操作を実行します。

- API Gateway、API Gateway UI およびダッシュボード (Kibana) をホストしている古い Integration Server の設定、および 1 つのAPI Gateway インスタンスの設定のバックアップを含む圧縮ファイルを作成します。1 つの API Gateway インスタンスのみ移行できます。
- Event Data Store のスナップショットの作成。Event Data Store は以下を含みます。
  - API
  - アプリケーション
  - ポリシーとポリシーアクション
  - 設定と管理設定
  - エイリアス
  - ユーザ、グループおよびアクセスプロファイル
  - プラン、パッケージおよびサブスクリプション
  - トランザクション用 API イベント、監視、ポリシー違反、脅威からの保護、ライフサイクルおよびエラー
  - パフォーマンスメトリクス

設定のバックアップと Event Data Store スナップショットを作成する方法については、次の手順に従ってください。

1. 古い API Gateway を起動します。
2. コマンドウィンドウまたはシェルを開いて、*old_Software AG_directory* /IntegrationServer/instances/*instance_name* /packages/WmAPIGateway/cli/bin ディレクトリに移動します。*instance_name* は移行する API Gateway インスタンスです。次に、以下のコマンドを実行します。

   backupDestinationDirectory パラメータは、設定のバックアップを含む圧縮ファイルと Event Data Store スナップショットが保存されるディレクトリです。ディレクトリはすでに存在している必要があり、書き込み権限を持っている必要があります。スナップショットは default という名前のサブディレクトリに保管されます。圧縮ファイルとスナップショットは、それぞれ apigw_upgrade_backup.zip と apigw_upgrade_backup という名前で保存されます。

   ```
   apigw-upgrade-backup.[bat|sh]
   -backupDestinationDirectory full_path_to_directory_for_
   configuration_backups_ZIP_file_and_EDS_snapshot
   ```

3. 新旧の API Gateway インストールが異なるマシン上にある場合は、`-backupDestinationDirectory` を新しいマシンにコピーします。

古い API Portal の準備

1. 古い API Portal Cloud Controller を起動します。
2. 古い API Portal がクラスタ化された高可用性設定でインストールされている場合は、すべてのノードが実行中で、ZooKeeper アンサンブルにアクセス可能であることを確認します。次のコマンドを実行して、各ノードを親ノードに登録します。

   ```
   acc> add node logical_node_name IP_address_or_host_name
   [@port ] user_name password
   ```

3. 次のコマンドを実行します。

   ```
   acc> startall
   ```

4. 次のコマンドを実行して、すべての API Portal コンポーネントが実行されていることを確認します。

   ```
   acc> list
   ```

5. 古い API Portal のテナントごとに、新しい API Portal 内にテナントを作成する必要があります。次のコマンドを実行して、古い API Portal 内のすべてのテナントをリストします。

   ```
   acc> list tenants
   ```

6. 古いテナントごとに API および関連データ、ユーザデータ、API Portal ドキュメント記憶領域データ (すべてのアクセス権限を含む)、コラボレーションデータをファイルにバックアップします。各ファイルにはファイル拡張子 .acb が自動的に追加されます。

   以下の表は、古いテナントのデータをそれぞれの古い製品リリースのファイルへバックアップする方法を説明しています。

古いリリース	実行するコマンド
9.7	acc> set acc config backup.restore.tenant.app.types = com.aris.umcadmin.y-umcadmin-run-prod, com.aris.modeling.components.y-server-run-prod, com.aris.adsadmin.y-adsadmin-run-prod, com.aris.runnables.ecp-run-prod acc> set acc config backup.restore.tenant.app.type.names = umc, abs, ads, ecp acc> backup tenant tenant_name to full_path_to_backup_file username=your_user_name password=your_password
9.8 9.9、9.10、9.12	acc> set acc config backup.restore.tenant.app.types=UMC,ABS,ADS,ECP acc> backup tenant tenant_name to full_path_to_backup_file username=your_user_name password=your_password
10.1 10.2	acc> backup tenant tenant_name to full_path_to_backup_file username=your_user_name password=your_password

7. 古い API Portal のカスタマイズされた表示を作成していた場合は、次のように表示をバックアップします。
   a. ブラウザで API Portal を開き、Administrator のクレデンシャルでログオンします。

b. [**Administration**] ページに移動します。[**Customization**] セクションの [**Views**] ページで、カスタム表示の名前にマウスカーソルを重ね、[**Backup**] をクリックします。API Portal によって、カスタマイズされた表示を含む圧縮ファイルが作成されます。

c. 圧縮ファイルを保存します。

古い CentraSite の準備

1. CentraSite Control の設定をデフォルトから変更した場合は、設定をメモに書きとめ、新しい CentraSite インストールで同様の変更を加えることができるようにします。
2. 新旧の CentraSite が同じマシン上にある場合、古い CentraSite がシャットダウンされていることを確認してください。
3. 古い CentraSite の設定データと、古いレジストリリポジトリのアセットを、圧縮ファイルにエクスポートします。古いマシンでコマンドウィンドウまたはシェルを開いて、*old_Software AG_directory* /CentraSite/utilities ディレクトリに移動し、適切なコマンドを実行します。

   - Windows では、`sbsExport.cmd full_path_to_ZIP_file` を実行する
   - UNIX, では、`sbsExport.sh full_path_to_ZIP_file` を実行する

   UNIX コマンドの例:

   ```
   ./sbsExport.sh /tmp/sbs_cs82_data.zip
   ```

4. 新旧の CentraSite インストールが異なるマシン上にある場合は、圧縮ファイルを新しい CentraSite をホストするマシン上の任意のディレクトリにコピーします。

古い Deployer の準備

古いホストの Integration Server を起動します。古い Deployer で、新しいソース/ターゲット製品サーバをポイントするように、古いソース/ターゲット製品サーバへの接続をすべて編集します。ユーザ名とパスワードが不明な場合、これらを入力する必要はありません。手順については、『*webMethods Deployer User's Guide*』を参照してください。

古い Software AG Designer の準備

Integration Server または Microservices Runtime 定義のエクスポート

1. 古い Software AG Designer で、[**Window**] > [Preferences] を選択します。[**Preferences**] ダイアログボックスにある左側のナビゲーションバーで、[**Software AG**] > [Integration Server] を選択します。
2. [**エクスポート**] をクリックして、ダイアログボックスを完了します。Software AG Designer により、拡張子 .properties でファイルが保存されます。

CloudStreams のサーバ定義とプロジェクトのエクスポート

1. CloudStreams のサーバ定義をファイルにエクスポートするには、[Window] > [Preferences] > [Software AG] > [**CloudStreams** サーバ] に移動します。[エクスポート] をクリックして、ダイアログボックスを完了します。Software AG Designer により、拡張子 .properties でファイルが保存されます。
2. [CloudStreams 開発] パースペクティブを開きます。

   CloudStreams Governance ビューでエクスポートする Governance プロジェクトを右クリックし、[Export] をクリックしてウィザードを完了します。プロジェクトフォルダがエクスポートされます。

   CloudStreams プロバイダビューでエクスポートするプロバイダプロジェクトを右クリックし、[Export] をクリックしてウィザードを完了します。プロジェクトがアーカイブ (圧縮ファイル) 形式でエクスポートされます。

設定のエクスポート

設定を新しい Software AG Designer に移行する場合は、古い Software AG Designer からエクスポートして新しい Software AG Designer にインポートします。設定をエクスポートするには、以下の手順に従います。

1. 古い Software AG Designer で、[**File**] > [Export] ウィザードを開きます。[Select] パネルで、[**General**] > [Preferences] を選択して、[**Next**] をクリックします。
2. [**To preference file**] フィールドに、設定のエクスポート先とするファイルの名前を指定します。Software AG Designer により、拡張子 .epf でファイルが保存されます。

古い Integration Server または Microservices Runtime の準備

メモ: Microservices Runtime の手順は Integration Server の場合と同じです。

製品の起動と接続

1. 古い Integration Server を起動し、古い Integration Server Administrator を開きます。
2. Universal Messaging を使用している場合は、Integration Server が、webMethods メッセージングプロバイダとして機能している各 Universal Messaging サーバに接続されていることを確認します。別の JMS プロバイダを使用している場合は、Integration Server が JMS プロバイダ に接続されていることを確認します。

トリガーを一時停止してキューを空にする

9.8 からのアップグレード: Command Central を使用してアップグレードする際にトリガーを一時停止してキューを空にする

Command Central から Integration Server インスタンスを休止することで、トリガーを一時停止してメッセージキューを空にすることができます。

Command Central を使用しないでアップグレードする際にトリガーを一時停止してキューを空にする

1. [設定] > [休止] ページに移動します。[休止モードを開始する] をクリックして休止を開始する時間を少なくとも 1 分に設定します。これは、Integration Server が新しく受信する要求の実行を停止し、実行中のサービスを完了する時間を確保するためです。休止レポートのすべてのフィールドで [状態] が「SUCCESS」と表示されていることを確認してください。Integration Server が休止された場合に発生する特定のアクションの詳細については、『*webMethods Integration Server Administrator's Guide*』を参照してください。
2. [**設定**] > [メッセージング] > [webMethods トリガーの管理] ページに移動します。[現在のキューカウント] フィールドに各トリガーで 0 が表示されない場合は、診断して問題を修正します (webMethods メッセージングプロバイダがアクティブでない、Integration Server からの要求の処理が遅いなど)。すべてのトリガーの [**現在のキューカウント**] フィールドに 0 が表示されるまで、ページを更新します。
3. [**設定**] > [メッセージング] > [webMethods メッセージングの設定] ページに移動します。Universal Messaging 接続エイリアスの [CSQ カウント] フィールドに 0 が表示されていることを確認します。
4. [**設定**] > [メッセージング] > [JMS 設定] ページに移動します。[**JMS 接続エイリアスの定義**] 領域で、すべての JMS 接続エイリアスに対して、[**CSQ カウント**] フィールドが 0 になっていることを確認します。

古い MashZone NextGen の準備

9.10 または 9.12 からのアップグレード: MashZone NextGen 10.0 以降は、PPM チャートビューをサポートしていません。MashZone NextGen データベースからビューを削除するには、MashZone NextGen Developer または MashZone NextGen Administrator を開き、API Console に移動して、次の要求を送信します。

```
{
  "version": "1.1",
  "sid": "AppService",
  "oid": "removeApp",
  "svcVersion": "0.1",
  "params": [
    "ppm-chart"
  ]
}
```

true の応答は、ビューが削除されたことを示します。

古い My webMethods Server の準備

My webMethods Server の設定を編集する場合は、MywebMethodsServer データベースコンポーネントから My webMethods Server インストールに適切な設定ファイルをダウンロードしてから、ファイルを変更します。次に、ファイルをデータベースコンポーネントにアップロードしてファイルシステムから削除するか、ファイルシステムに保存してその設定がデータベースの同等の設定よりも優先して使用されるようにします。*old_Software AG_directory* /MWS/server/*instance_name* /config ディレクトリに移動し

て、そのようなファイルがあるかどうかを確認します。新しいインストールに移行する必要がないファイルは、古いインストールから削除します。

古い OneData の準備

1. 古い OneData で、[**ホーム**] > [管理者] > [ジョブセンター] ページに移動します。[**Filter**] ページで、各ジョブタイプに対して [**Active**] または [**Pending Active**] でフィルタします。ジョブがこれらのいずれかの状態である場合は、ジョブを終了するか、完了するまで待機します。
2. OneData で JMS を使用している場合は、*old_Software AG_directory* /profiles/ODE/webapp/onedata/WEB-INF/lib ディレクトリに移動して、JMS プロバイダに必要なすべてのクライアント jar ファイルをバックアップします。
3. OneData でケルベロス認証を使用している場合は、ケルベロスベースの SSO 設定およびファイルをバックアップします。

   以下の表は、ケルベロスベースの SSO 設定とバックアップするファイルを含むディレクトリを一覧しています。

old_Software AG_directory /	バックアップするファイル
profiles/ODE/configuration/tomcat/conf	server.xml
profiles/ODE/configuration	custom_wrapper.conf および jaas.config
profiles/ODE/workspace/webapps/onedata/WEB-INF	web.xml
profiles/ODE/configuration/com.softwareag.platform.config.propsloader	すべての com.softwareag.jaas.realm.pid-*SSO_realm_name*

古い Optimize の準備

1. CCS (Central Configurator System) 環境定義を再利用したい場合は、環境を XML ファイルにエクスポートします。My webMethods で、[**管理**] > [システム全体] > [環境] > [環境の定義] ページに移動します。環境の隣にあるチェックボックスを選択し、[**環境のエクスポート**] をクリックし、エクスポートする環境のファイル名と場所を指定します。
2. 新しいインストールで Analytic Engine をクラスタ化する場合は、Terracotta Server Array が必要です。計画については、『 *Using Terracotta with webMethods Products*』、『*Configuring BAM*』および Terracotta BigMemory Max のマニュアルを参照してください。

古い Zementis Predictive Analytics の準備

1. *old_Software AG_directory* /Zementis/adapa-app/migration ディレクトリに移動します。
2. 次のコマンドを実行して、Zementis Predictive Analytics モデルおよびリソースをバックアップします:

   ./migration.sh backup Zementis_host_URL admin_user admin_password

   バックアップが adapa-backup-*yymmdd -hhmmss* ディレクトリに書き込まれます。*yymmdd -hhmmss* は、UTC タイムゾーンに基づいてコマンドを実行した時間です。

古い製品のシャットダウン

Command Central を使用した古い製品のシャットダウン

Command Central から製品をシャットダウンします。

古い製品を手動でシャットダウン

以下の表は、特定の製品をシャットダウンする方法を一覧しています。

製品	シャットダウンの手順
API Portal	古い API Portal と新しいものが同じマシン上にある場合は、次の手順に従います。 ■ 古い API Portal Cloud Controller でこのコマンドを実行します。 acc> stopall ■ *new_Software AG_directory* /API_Portal/server/bin ディレクトリに移動し、以下のコマンドを実行して、クラウドエージェントを停止します。 CloudAgentApp.{bat|sh} stop
CentraSite	新旧の CentraSite が同じマシン上にある場合は、Software AG Runtime サービスを停止して、すべての古い Application Server Tier をシャットダウンし、古いレジストリリポジトリのサービスを停止して、シャットダウンします。
EntireX	上記の表の説明どおりに EntireX をシャットダウンしてから、EntireX ライブラリを使用するすべての Broker、RPC サーバ、顧客アプリケーションをシャットダウンします。手順については、古いリリース用の製品マニュアルを参照してください。
Terracotta	webMethods 以外のすべてのクライアントをシャットダウンします。

Windows システムのその他の製品をシャットダウンするには

- 製品がサービスとして実行されている場合は、Windows のサービスウィンドウからシャットダウンします。サービスは Software AG *product release* として表示されます。
- 製品がアプリケーションとして実行されている場合は、Windows のスタートメニューからシャットダウンします。アプリケーションは、[**Software AG**] > [サーバの停止] > *product* として表示されます。

UNIX システムの場合は、古いリリース用の製品マニュアルの手順を使用します。

メモ: 移行しようとしているデータベースコンポーネントに接続している、すべての古い ActiveTransfer、Integration Server、Microservices Runtime、My webMethods Server、OneData および Optimize インスタンスをシャットダウンしておく必要があります。

移行するデータベースコンポーネントの準備

概要

新しいリリースでは、新しいテーブル、列、キー、インデックスなど、データベースコンポーネントの変更が必要になる場合があります。この節では、新しい製品リリースと互換性を持つように既存のデータベーススキーマを更新するデータベース移行スクリプトを実行します。スクリプトは、既存のデータベースコンポーネントを変更するか、新しい構造を持つ並行データベースコンポーネントを作成して、テーブル、列、キー、インデックスを INSERT、SELECT、RENAME および DROP します。これらの変更で、データベースのサイズが増える場合があります。

データベースコンポーネントの移行に関する特別な考慮事項

- ビジネスプロセスを使用している場合は、処理中、完了、失敗などのビジネスプロセスインスタンスに関する情報が保存されます。この保存された情報は、ビジネスプロセスのランタイムデータと呼ばれます。新しいインストールで、古いインストールからの処理中のプロセスインスタンスの実行を完了しない場合、または完了または失敗したプロセスインスタンスを再サブミットしない場合は、ビジネスプロセスのランタイムデータを移行しないでください。
- Optimize で DB2 RDBMS を使用している場合、データベースコンポーネント移行コマンドで指定する URL には特別な考慮事項があります。以下の点に従う必要があります。
  - URL のスキーマ名はすべて大文字で指定します。
  - オプション `CreateDefaultPackage=true,ReplacePackage=true` および `DynamicSections=3000` を指定します。この設定は、同じスキーマまたはデータベースにあるすべてのデータベースコンポーネントに影響します。

Trading Networks データベースコンポーネントを移行する前に

Trading Networks データベースコンポーネントに対してカスタムインデックスを作成した場合は、これらのカスタムインデックスが、次の手順で Trading Networks データベース移行スクリプトを実行するとき

に作成されるインデックスと競合するかどうかを確認します。競合する場合は、カスタムインデックスを破棄してください。

以下の表は、各リリースの Trading Networks 用のデータベース移行スクリプトが入っている *new_Software AG_directory* /common/db/scripts/*RDBMS* /tradingnetworks/migrate ディレクトリ下のディレクトリを一覧しています。

古いリリース	ディレクトリ
9.7	/50_to_55
9.8	/55_to_56 および 56_to_65
9.9	/65_to_70

データベース移行スクリプトへの修正のインストール

データベースコンポーネントの移行に Command Central を使用している場合は、*Software AG Command Central Help* の「*Before Creating Database Components*」の手順に従って、データベース移行スクリプトに最新の修正をインストールします。

データベースコンポーネントの移行に Database Component Configurator を使用している場合は、Software AG Update Manager を使用してデータベース移行スクリプトに最新の修正をインストールします。

データベーススクリプトの修正名は、DC_*old_release_number* _DBS_Fix*latest_fix_number* の規則に従っています。

移行する古い製品インストールのソースとして圧縮ファイルの作成

新旧の製品インストールが異なるマシン上にある場合、古い製品インストールの圧縮ファイルを作成して、移行ソースとして使用できます。

Command Central を使用して移行する場合の圧縮ファイルの作成

Command Central を使用してある環境をアップグレードする場合、Command Central から古いインストールの圧縮ファイルを作成するか、または次の項で説明するような独自の圧縮ファイルを作成することができます。

コンポジットテンプレートを使用してアップグレードを自動化する場合は、Command Central によって古いインストールディレクトリごとに圧縮ファイルが自動的に作成されます。圧縮ファイルには、.log ファイル以外のすべてのファイルが含まれています。圧縮ファイルのサイズを縮小したい場合は、それぞれ

の製品で必要なファイルだけを圧縮ファイルに含めるコマンドの表が次の項で説明されていますので、参照してください。

移行ユーティリティを使用して移行する場合の圧縮ファイルの作成

My webMethods Server 用の圧縮ファイルの作成

1. 古いマシンで、コマンドウィンドウまたはシェルを開き、*old_Software AG_directory* /MWS/bin/migrate または *old_Software AG_directory* /MWS/bin ディレクトリ (所有している現在のリリースによって異なる) に移動し、コマンド `zip-mws.{bat|sh}` を実行します。mws.zip という名前の圧縮ファイルが *old_Software AG_directory* /MWS/migrate-zip ディレクトリに作成されます。
2. mws.zip ファイルを、新しい My webMethods Server をホストするマシンの任意のディレクトリにコピーします。

   重要: FTP を使用してコピーする場合は、バイナリファイル転送モード/タイプを使用してください。他のモード/タイプを使用すると、圧縮ファイルが壊れることがあります。

その他すべての製品用の圧縮ファイルの作成

以降の手順では、Java Archive ツールを使用して圧縮ファイルを作成します。古い製品インストールをホストするマシンで JAVA_HOME および PATH システム変数に Java Archive ツールの場所を指定します。ツールは *Software AG_directory* /jvm/jvm/bin ディレクトリにあります。

メモ: 一部のシステムでは、下位レベルの jvm ディレクトリ名に、/jvm/jvm160_32、/jvm/jvm170、/jvm/jvm_64 など、追加的な情報が含まれます。

1. 製品の古いマシンに移動し、コマンドウィンドウまたはシェルを開きます。
2. 古い製品を含む Software AG ディレクトリに移動し、以下の表に示されたコマンドを入力します。複数のコマンドが記載されている場合は、各コマンドで同じ圧縮ファイル名を使用します。
3. 製品の圧縮ファイルを、新しい製品をホストするマシンの任意のディレクトリにコピーします。

   重要: FTP を使用してコピーする場合は、バイナリファイル転送モード/タイプを使用してください。他のモード/タイプを使用すると、圧縮ファイルが壊れることがあります。

以下の表は、移行のソースとして使用する古い製品インストールの圧縮ファイルを作成するためのコマンドを一覧しています。

製品	実行するコマンド
Application Platform	最初にログファイルを *old_Software AG_directory* /profiles/*product* /logs ディレクトリから移動することで、圧縮ファイルのサイズを小さくできます。*product* はホストの Integration Server インスタンス、My webMethods Server インスタンス、または SPM (Platform Manager) です。 `jar cfM ZIP_file_name .zip install/products profiles`

製品	実行するコマンド
Digital Event Services	インストールの *new_Software AG_directory* /common/migrate/DigitalEventServices/bin ディレクトリに migrate.{bat|sh} ファイルが含まれている場合は、次のコマンドを実行します。 jar cfM ZIP_file_name .zip install/products jar ufM ZIP_file_name .zip common/DigitalEventServices 次のディレクトリが存在する場合は、ディレクトリを圧縮ファイルに追加します。 jar ufM ZIP_file_name .zip profiles/product /configuration/DigitalEventServices
Event Routing	インストールの *new_Software AG_directory* /common/migrate/EDA/bin ディレクトリに migrate.{bat|sh} ファイルが含まれている場合は、次のコマンドを実行します。 jar cfM ZIP_file_name .zip install/products jar ufM ZIP_file_name .zip common/conf common/EventTypeStore 次のディレクトリが存在する場合は、ディレクトリを圧縮ファイルに追加します。 jar ufM ZIP_file_name .zip profiles/product /configuration /event/routing optimize/analysis/conf/event/routing
Software AG Infrastructure (特に Software AG Runtime)	最初にログファイルを *old_Software AG_directory* /profiles/CTP/logs ディレクトリから外に移動することで、圧縮ファイルのサイズを小さくできます。 jar cfM ZIP_file_name .zip common/conf install/products profiles/CTP 「profiles/CTP: no such file or directory」というメッセージが表示されることがあります。このメッセージは無視できます。
Infrastructure Data Collector	最初にログファイルを *old_Software AG_directory* /profiles/InfraDC/logs ディレクトリから移動することで、圧縮ファイルのサイズを小さくできます。 jar cfM ZIP_file_name .zip common/conf install/products profiles/InfraDC
Integration Server または Microservices Runtime	最初にログファイルを *old_Software AG_directory* /profiles/*instance_name* /logs、/IntegrationServer/instances/logs、および /IntegrationServer/instances/*instance_name* /logs ディレクトリから移動することで、圧縮ファイルのサイズを小さくできます。 jar cfM ZIP_file_name .zip *
MashZone NextGen	jar cfM mzng.zip MashZoneNG
OneData	jar cfM ZIP_file_name .zip install/products profiles/ODE/bin/onedata/config/*

製品	実行するコマンド
Universal Messaging	jar cfM ZIP_file_name .zip install/products UniversalMessaging/server

6 データベースコンポーネントの移行

- 注意 48
- Command Central を使用したデータベースコンポーネントの移行 48
- Database Component Configurator を使用したデータベースコンポーネントの移行 48
- DynamicBusinessOrchestrator データベースコンポーネントの移行 49

注意

重要: データベースコンポーネントを新しいリリースに移行すると、これらのコンポーネントは古い環境で使用できなくなります。

Command Central を使用したデータベースコンポーネントの移行

データベースコンポーネントを移行するには、 Command Central コマンドの `sagcc exec administration` を使用します。コマンド構文は以下のように表示されます。パラメータは、Database Component Configurator を実行する際に使用するパラメータと同じであり、『*Installing Software AG Products*』に記載されています。

```
sagcc exec administration product DCC_node_alias DatabaseComponentConfigurator
database migrate [parm1=value1 ] [parm2=value2 ]...
```

次の例では、データベースコンポーネントが jdbc:wm:sqlserver://DBserver:1433;databaseName=TESTDB の SQL Server RDBMS 上にあり、データベースユーザは webmuser、パスワードは webmpass であることを示しています。

- Integration Server データベースコンポーネントを 9.9 から最新のリリースにアップグレードするには:

  ```
  sagcc exec administration product local DatabaseComponentConfigurator
  database migrate db.type=sqlserver product=IS version=latest
  db.username=webmuser db.password=webmpass db.name=TESTDB
  db.url="jdbc:wm:sqlserver://DBserver:1433;databaseName=TESTDB"
  ```

- My webMethods Server データベースコンポーネントを最新のリリースにアップグレードするには:

  ```
  sagcc exec administration product local DatabaseComponentConfigurator
  database migrate db.type=sqlserver component=MWS version=latest
  db.username=webmuser db.password=webmpass db.name=TESTDB
  db.url="jdbc:wm:sqlserver://DBserver:1433;databaseName=TESTDB"
  ```

Command Central で移行の進捗を確認するには、Command Central Web ユーザインタフェースで [Jobs] をクリックします。

Database Component Configurator を使用したデータベースコンポーネントの移行

新しい Database Configuration をインストールしたマシンでコマンドウィンドウまたはシェルを開き、*new_Software AG_directory* /common/db/bin ディレクトリに移動し、以下のコマンドを実行してデータベースコンポーネントを移行します。

```
dbConfigurator.{bat|sh} -a migrate -d {oracle|sqlserver|db2luw}
{-c db_component | -pr product } -v latest -l db_server_URL
-u existing_db_user -p password
```

各コマンドを実行した後に、*new_Software AG_directory* /common/db/logs ディレクトリにあるログファイル dcc_*yyyymmddhhmmss* を確認します。

Oracle または DB2 RDBMS を使用しており、デフォルトのテーブルスペースを使用していない場合は、-tsdata data_tspace_nameおよび -tsindex index_tspace_nameパラメータも指定します。

AIX、HP-UX、または Solaris システムでは、値 “db_server_URL”、”existing_db_user” および “password” を引用符で囲みます。

アップグレードする製品	***product* の値**
Integration Server または Microservices Runtime	-pr IS **メモ:** 他のデータベースコンポーネントを移行する前に Integration Server または Microservices Runtime データベースコンポーネントを移行します。
ActiveTransfer Server	-c ActiveTransfer
API Gateway	-c APIGatewayEvents
ビジネスプロセスランタイムと Business Rules	-pr BPM
Archive、Staging、または Reporting データベースコンポーネント	-c {Archive|Staging|Reporting}
My webMethods Server	-pr MywebMethodsServer
OneData	-c {OneDataMetadata|OneDataReleaseArea|OneDataWorkArea}
Optimize	-pr OPTIMIZE
Trading Networks	-pr TN

DynamicBusinessOrchestrator データベースコンポーネントの移行

DynamicBusinessOrchestrator データベースコンポーネントに対する SQL ステートメントを実行します。

RDBM	
Oracle、DB2、MySQL	UPDATE PRA_PROCESS_RECENT P SET P.system = (SELECT DISTINCT PD.TYPE FROM WMPROCESSDEFINITION PD WHERE PD.PROCESSKEY = P.PROCESSKEY) UPDATE PRA_PROCESS_RECENT P SET P.system = (SELECT DISTINCT PD.TYPE FROM WMPROCESSDEFINITION PD WHERE PD.PROCESSKEY = P.PROCESSKEY)
SQL Server	UPDATE PRA_PROCESS SET PRA_PROCESS.system = WMPROCESSDEFINITION.TYPE FROM PRA_PROCESS, WMPROCESSDEFINITION WHERE WMPROCESSDEFINITION. PROCESSKEY = PRA_PROCESS.PROCESSKEY; UPDATE PRA_PROCESS_RECENT SET PRA_PROCESS_RECENT.system = WMPROCESSDEFINITION.TYPE FROM PRA_PROCESS_RECENT, WMPROCESSDEFINITION WHERE WMPROCESSDEFINITION. PROCESSKEY = PRA_PROCESS_RECENT.PROCESSKEY;

7 Software AG Infrastructure の移行

- 移行される Software AG Infrastructure の設定、データおよびアセット 52
- Software AG Infrastructure の移行 52
- Software AG Infrastructure のアップグレードの完了 53

移行される Software AG Infrastructure の設定、データおよびアセット

- Software AG Runtime 監視、デバッグ、JAAS、JMX、ポート、セキュリティ、シングルサインオン、ウォッチドッグ、および Web Services Stack 設定。
- ジャーナルログ設定、ユーザリポジトリ、およびパスワードストア。
- #include ディレクティブとコメント (関連付けられたプロパティは除く) を含む、古い custom_wrapper.conf ファイルに加えられた Java サービスラッパーのカスタマイズ内容。
- ユーザ、グループ、役割
- プロキシタイプ HTTP、HTTPS、FTP、および SOCKS のプロキシ設定ファイル。ユーティリティは、新しいインストールでまだ設定されていないプロキシのファイルを移行します。
- 10.0 以降では、Software AG Security Infrastructure は SSX または SSX LoginModule のサポートを終了しました。移行ユーティリティは、SSX LoginModule を含まない LoginContext を移行します。ただし、古い LoginContext に SSX LoginModule が含まれている場合は、移行ユーティリティは次のように処理します。
  - LoginContext が古いリリースと新しいリリースに存在する場合、ユーティリティは新しい LoginContext に移行が実行されていないというコメントを追加します。
  - LoginContext が古いリリースに存在して新しいリリースには存在しない場合、ユーティリティは新しいリリースに移行が実行されていないというコメントと共に空の LoginContext を追加します。これは、古い LoginContext を使用したソフトウェアが、コンテキスト不足のために失敗することを回避するためです。

10.1 以降、移行ユーティリティは Software AG インストールディレクトリにない JKS ファイルの移行をサポートしません。

Software AG Infrastructure の移行

Infrastructure コンポーネントは、製品と共に自動的にインストールされます。Infrastructure コンポーネントは、Software AG Common Platform、Software AG Runtime、Software AG Security Infrastructure および Software AG Web Services Stack です。

1. 実行されている新しい製品がないことを確認します。新旧の製品が同じマシン上にある場合は、古い製品がシャットダウンされていることを確認します。
2. 移行ユーティリティを使用して移行する場合は、123 ページの「移行ユーティリティを使用した製品の移行」の手順に従ってください。Software AG Infrastructure は Command Central を使用してアップグレードできません。

Software AG Infrastructure のアップグレードの完了

- *old_Software AG_directory* /profiles/CTP/configuration/com.softwareag.platform.config.propsloader ディレクトリに移動し、すべてのファイルを *new_Software AG_directory* /profiles/CTP/configuration/com.softwareag.platform.config.propsloader ディレクトリにコピーします。
- 新しい custom_wrapper.conf ファイルの最後に #include ディレクティブが移行されます。それらを確認し、必要に応じて調整します。custom_wrapper.conf ファイルは *Software AG_directory* /profiles/CTP/configuration ディレクトリにあります。

  メモ: 『*Software AG Infrastructure Administrator's Guide*』に記載されているように、Software AG から指示がない限り、wrapper.conf ファイルを変更しないでください。wrapper.conf ファイルを変更した場合、古い wrapper.conf ファイルで変更したプロパティの値を、新しい custom_wrapper.conf ファイルの対応するプロパティに手動でコピーします。

8 Terracotta の移行

- Terracotta BigMemory の移行 56
- Terracotta DB の移行 56

Terracotta BigMemory の移行

この節では、webMethods 製品のサポートで展開されている Terracotta BigMemory Max クライアントおよびサーバを移行する方法について説明します。その他のすべての設定については、Terracotta BigMemory Max のマニュアルを参照してください。

『*Using Terracotta with webMethods Products*』の手順に従って、4.1.4 以降のライセンスキーと tc-config.xml ファイルをインストールします。

クラスタを再起動する前に、クラスタ内のすべての Terracotta BigMemory Max クライアントおよびサーバを、新しいリリースにアップグレードする必要があります。

サーバが 1 台の場合

1. 古い Terracotta Server Array をシャットダウンします。
2. 新しい Terracotta Server Array を開始します。

アクティブサーバおよびミラーサーバで構成されるミラーグループがある場合

1. 古いミラーサーバをシャットダウンします。
2. 古いアクティブサーバをシャットダウンします。
3. 新しいアクティブサーバを起動します。
4. 新しいミラーサーバを起動します。

Terracotta DB の移行

Terracotta DB 10.3 は、クラスタトポロジやデータ記憶領域の形式、その他の機能の操作の点で、Terracotta BigMemory Max 4.x および 3.7.6 とは大きく異なります。Terracotta BigMemory Max サーバのデータと設定を Terracotta DB サーバへ移行することはできません。TCStore データは移行できません。

Terracotta Management Console 設定とデータの移行

1. *old_Software AG_directory* /TerracottaDB/tools/management/conf ディレクトリに移動して、tmc.properties ファイルを新しいインストールディレクトリの同じ場所にコピーします。
2. *old_Software AG_directory* /TerracottaDB/tools/management/conf ディレクトリに移動して、データディレクトリを新しいインストールディレクトリの同じ場所にコピーします。
3. Terracotta Management Console を開始します。
4. エラーメッセージを受け取った場合は、tmc.properties ファイルを開いて、すべてのプロパティが正しく設定されていることを確認してください。手順については、製品マニュアルを参照してください。

Ehcache データの移行

再起動可能なキャッシュマネージャに含まれる再起動可能なキャッシュ内の Ehcache データは、プラットフォームのデータルートを移動せずに新しいクラスタに移行できます。このデータの移行はデータディレクトリレベルで動作するため、同じデータディレクトリを使用するすべての再起動可能なキャッシュマネージャのデータが一緒に移行されます。

1. 古いクラスタをシャットダウンします。
2. 古いクラスタと同じ数のストライプと共に新しいクラスタ内でアクティブサーバを起動します。
3. クライアントを使用してキャッシュマネージャ設定を作成します。キャッシュマネージャのクラスタティアマネージャ名を含むクラスタ URI は古いクラスタの URI と異なるものにできます。URI の名前が異なる場合、キャッシュマネージャ設定 API を使用するときに、古い名前を再起動識別子として指定してください。これにより、システムは指定されたキャッシュマネージャに対応するデータを正しくマップすることができます。同じデータディレクトリ下に複数のキャッシュマネージャが存在する場合、設定 API を使用して、ターゲットクラスタ内にすべてのキャッシュマネージャを作成します。手順については、『Ehcache API Developer Guide』の「Fast Restartability」についてのセクションを参照してください。
4. 新しいクラスタをシャットダウンします。
5. 移行するすべてのサーバ上のすべての再起動可能なキャッシュマネージャに対して、Ehcache ディレクトリ （キャッシュデータルート） のコンテンツを、古いクラスタのすべてのアクティブサーバのデータディレクトリから新しいクラスタのデータディレクトリにコピーします。新しいクラスタのデータディレクトリパスは、古いクラスタのデータディレクトリと異なるものにできます。
6. 新しいクラスタを開始します。Terracotta DB は、古いクラスタから移行されたすべてのキャッシュデータをロードします。

9 Universal Messaging の移行

- 移行できる Universal Messaging 設定、データおよびアセット 60
- Universal Messaging の移行 60
- Universal Messaging のアップグレードの完了 60

移行できる Universal Messaging 設定、データおよびアセット

- サーバインスタンス。
- サーバインスタンスデータのディレクトリ。

Universal Messaging の移行

1. Universal Messaging クラスタをアップグレードする場合、Universal Messaging のマニュアルでクラスタリングの手順を参照してください。
2. 新旧の Universal Messaging インストールが同じマシン上にある場合、古いインストールがシャットダウンされていることを確認してください。
3. 次のいずれかの方法で移行できます。
   - 移行ユーティリティを使用して移行する場合は、123 ページの「移行ユーティリティを使用した製品の移行」の手順に従ってください。
   - Command Central を使用して移行する場合は、以下の Command Central コマンドを使用してください。カスタム移行を使用する場合は、最初に移行ユーティリティを手動で実行して migrate.dat ファイルを作成し (手順については 123 ページの「移行ユーティリティを使用した製品の移行」 を参照)、`importFile` パラメータで migrate.dat ファイルを指定します。すべてのインスタンスを移行する場合は、`instanceName` オプションを指定しないでください。

     ```
     sagcc exec administration product target_node_alias
     NumRealmServer migration migrate
     {srcDir|srcFile}=full_path_to_{old_Software AG_directory|ZIP file}
     [importFile=full_path_to_migrate.dat ]
     [instanceName=name [,name ...]]
     ```

   新しいインストールで Template Applications および Enterprise Manager をインストールした場合は、移行で Template Applications と Enterprise Managerのインスタンスが作成されます。

Universal Messaging のアップグレードの完了

1. リリース 9.8 では、nserver.conf および nserverdaemon.conf ファイルが Server_Common.conf という単一ファイルに結合されました。Server_Common.conf ファイルは、*new_Software AG_directory* /UniversalMessaging/server/umserver/bin ディレクトリに保存されます。
   - 9.7 からのアップグレード: 古いインスタンスの nserver.conf または nserverdaemon.conf ファイルをカスタマイズした場合、対応する新しいインスタンスの Server_Common.conf ファイルにも同じ変更を行ってください。nserver.conf ファイルと nserverdaemon.conf ファイルは、*old_Software AG_directory* /nirvana/server/umserver/bin ディレクトリに保存されています。

- 9.8、9.9、または 9.10 からのアップグレード: 古いインスタンスの Server_Common.conf ファイルをカスタマイズした場合、対応する新しいインスタンスの Custom_Server_Common.conf ファイルにも同じ変更を行ってください。
- 9.12 または 10.1 からのアップグレード: 古いインスタンスの Server_Common.conf または Custom_Server_Common.conf ファイルをカスタマイズした場合、対応する新しいインスタンスの Custom_Server_Common.conf ファイルにも同じ変更を行ってください。

2. 古いインスタンスのデータディレクトリが Software AG Installer で指定されたデフォルトの場所にない場合、移行ユーティリティはデータディレクトリを移行していません。データディレクトリのデフォルトの場所は、*new_Software AG_directory* /UniversalMessaging/server/*instance_name* /data ディレクトリです。古い場所の古いデータディレクトリを使用するか、それを新しい場所にコピーすることができます。コピーした場合は、新しいインスタンスの Server_Common.conf ファイルの DDATADIR パラメータを編集し、新しいデータディレクトリの場所をポイントするように設定します。
3. 古いデフォルト JAAS 設定ファイルを変更した場合は、*old_Software AG_directory* /UniversalMessaging/server/*server_name* /bin ディレクトリに移動し、jaas.conf ファイルを開いて、新しいインストールの同じファイルに変更をコピーします。また、トラストストアファイルなどの外部リソースファイルへの参照をコピーし、必要に応じて参照を変更してください。

10 My webMethods Server の移行

- 移行できる My webMethods Server 設定、データおよびアセット 64
- 移行の一部として実行されるその他のアクション 64
- 埋め込みデータベースのサポート停止 64
- My webMethods Server の移行 64
- My webMethods Server のアップグレードの完了 65
- Business Rules のアップグレードの完了 66

移行できる My webMethods Server 設定、データおよびアセット

- サーバインスタンス。
- Business Console のユーザ設定、AppSpace、およびガジェット (古い My webMethods Server が Business Console をホストしていた場合)。
- #include ディレクティブとコメント (関連付けられたプロパティは除く) を含む、古い custom_wrapper.conf ファイルに加えられた Java サービスラッパーのカスタマイズ内容。

移行の一部として実行されるその他のアクション

- 新しい My webMethods Server で使用されない古い設定ファイルを削除し、新しい My webMethods Server で使用されない古いプロパティを新しい設定ファイルから削除します。
- My webMethods Server インストールディレクトリをデータベースと同期します。
- Universal Messaging の URL を指定した場合は、既存の JNDI URL パラメータを更新するか、必要に応じて cluster.xml ファイルに新しい JNDI URL パラメータを追加します。

埋め込みデータベースのサポート停止

My webMethods Server 10.3 から、My webMethods Server では埋め込みデータベースをサポートしていません。My webMethods Server 移行ユーティリティは、埋め込みデータベースを使用していた MWS インスタンスを移行しません。代わりに、新しいインスタンスを作成し、その後、Deployer または、My webMethods Server. のコンテンツのインポート/エクスポート機能を使用して、古いインスタンスから新しいインスタンスへアプリケーションやコンテンツを転送します。手順については、それぞれ *webMethods Deployer User's Guide* または *Administering My webMethods Server* を参照してください。

My webMethods Server の移行

1. My webMethods Server クラスタをアップグレードする場合、24 ページの「My webMethods Server の要件」を参照してください。
2. 新旧の My webMethods Server が同じマシン上にある場合、古い My webMethods Server がシャットダウンされていることを確認してください。
3. 次のいずれかの方法で移行できます。
   - 移行ユーティリティを使用して移行する場合は、123 ページの「移行ユーティリティを使用した製品の移行」の手順に従ってください。

- Command Central を使用して移行する場合は、以下の Command Central コマンドを使用してください。カスタム移行を使用する場合は、最初に移行ユーティリティを手動で実行して migrate.dat ファイルを作成し (手順については 123 ページの「移行ユーティリティを使用した製品の移行」 を参照)、importFile パラメータで migrate.dat ファイルを指定します。すべてのインスタンスを移行する場合は、instanceName オプションを指定しないでください。

  ```
  sagcc exec administration product target_node_alias
  MwsProgramFiles migration migrate
  {srcDir|srcFile}=full_path_to_{old_Software AG_directory|ZIP file}
  [importFile=full_path_to_migrate.dat ]
  [instanceName=name [,name ...]]
  [newNodeName=instance_name :new_node_name [,instance_name :new_node_name ...]]
  [cloneDbURL=URL  cloneDbUser=user  cloneDbPassword=password ]
  ```

My webMethods Server のアップグレードの完了

Java サービスラッパー #include ディレクティブの確認

移行ユーティリティは新しい custom_wrapper.conf ファイルの最後に #include ディレクティブを移行して確認し、必要に応じて調整します。custom_wrapper.conf ファイルは *Software AG_directory* /profiles/MWS_*instance_name* /configuration ディレクトリにあります。

メモ: 『*Software AG Infrastructure Administrator's Guide*』に記載されているように、Software AG の指示がない限り、wrapper.conf ファイルを変更しないでください。wrapper.conf ファイルを変更した場合、古い wrapper.conf ファイルで変更したプロパティの値を、新しい custom_wrapper.conf ファイルの対応するプロパティに手動でコピーします。

My webMethods Server インスタンスの初期化

初期化プロセスは、数時間かかる場合があります。

1. *new_Software AG_directory* /MWS/bin ディレクトリに移動し、以下のコマンドを実行して、新しい My webMethods Server サーバインスタンスを初期化します。

   ```
   mws.{bat|sh} -s instance_name  init
   ```

   新しいコンポーネントが展開され、インスタンスが自動的にシャットダウンします。インスタンスを再起動します。

2. 新旧の My webMethods Server インストールが異なるマシン上にある場合は、新しいインストールのホスト名を以下の手順で検証します。
   a. Administrator としていずれかの新しい My webMethods Server インスタンスにログオンし、[**管理**] > [My webMethods] > [クラスタ設定] > [MWS 用拡張 Web およびクラスタ設定] ページに移動します。
   b. [**ホスト**] および [**MWS Front End URL**] フィールドのホスト名が正しくない場合は、ホスト名を更新します。
   c. [**クラスタの状態および制御**] ページに移動し、インスタンスを再起動します。

クローンから稼働中のデータベースに切り替える

クローンデータベースを使用して新しいインストールをテストし (推奨)、後で稼働中のデータベースをテストした新しいインストールで使用する必要が生じた場合、My webMethods Server 移行ユーティリティをもう一度実行して切り替える必要があります。ユーティリティによって、My webMethods Server インストールディレクトリとデータベースが同期されます。

1. My webMethods Server 移行ユーティリティをもう一度実行します。
2. 各サーバインスタンスをもう一度移行します。各インスタンスで使用するデータベースを尋ねられたら、稼働中のデータベースを選択します。
3. 新しいインストール内の移行したインスタンスを削除するか保持するかを尋ねられたら、移行したインスタンスの削除を選択します。

Business Rules のアップグレードの完了

現在 My webMethods Server に展開された Business Rules プロジェクトは、最初にサーバインスタンスを開始したときに移行されます。クラスタ内で、開始する最初のクラスタノードがすべてのルールプロジェクトを移行します。移行中に、My webMethods Server に展開されているすべてのルールプロジェクトは使用できません。My webMethods Server ログファイルには、ルールプロジェクト移行についての詳細が含まれます。デフォルトでは、ログファイルは *Software AG_directory* /MWS/server/*instance_name* /logs ディレクトリに保管されます。

1. 新しいインストールディレクトリが古いインストールディレクトリと異なる名前である場合、Deployer が Business Rules アセット展開用に使用するファイルシステムの場所の妥当性検査を行い、必要に応じて My webMethods Server を再起動します。詳細については、『*Working with Business Rules in My webMethods*』の「Hot Deploying and Merging Rule Projects with webMethods Deployer」の「Configuring My webMethods Server」を参照してください。
2. ビジネス検証やデータプロバイダを使用する場合、新しい My webMethods Server インスタンスの 1 つに Administrator としてログインして、[管理] > [My webMethods] > [システムの設定内容] > [webMethods Business Rules 設定] ページに移動します。必要に応じて設定を更新し、変更を保存します。

   詳細については、『*Working with Business Rules in My webMethods*』を参照してください。ビジネス検証については、「Rule Verification Overview」の「Configuring a Server Connection for a PreconfiguredVerification」を参照してください。データプロバイダについては、「Working with Decision Tables」の「Configuring a Server Connection for a Pre-configured Data Provider Service」を参照してください。

11 Integration Server、Microservices Runtime、およびホストされている Wm パッケージの移行

- Integration Server およびMicroservices Runtime 68
- パッケージ 68
- 移行できる設定、データおよびアセット 68
- 移行の一部として実行されるその他のアクション 69
- Integration Server の移行 69
- Integration Server のアップグレードの完了 70
- ActiveTransfer のアップグレードの完了 72
- CloudStreams のアップグレードの完了 72
- Dynamic Business Orchestrator のアップグレードの完了 74
- アダプタのアップグレードの完了 74
- eStandards モジュールのアップグレードの完了 75

Integration Server およびMicroservices Runtime

この章は、Microservices Runtime および Integration Server の両方に適用されます。

パッケージ

このマニュアルでは、Integration Server の以下に示す各タイプのパッケージについて説明します。

- カスタムパッケージ。Software AG Designer でユーザが作成した Integration Server パッケージおよび Software AG Designer からユーザが生成したビジネスプロセスパッケージが含まれます。
- ホストされているパッケージ。製品選択ツリーの Integration Server の下のリストに表示される Software AG Installer の Software AG によって提供されるパッケージです。ツリーには、製品名を使用したリストが表示されます。ただし、ファイルシステム内および Integration Server 内では、Wm で始まるパッケージ名の下にリストが表示されます。このため、ホストされているパッケージは Wm パッケージとも呼ばれます。

  ホストされている Wm パッケージの例には、ActiveTransfer (WmMFT)、Process Engine (WmPRT)、Trading Networks (WmTN)、アダプタ、および eStandards モジュールがあります。

  新しい Integration Server をインストールした場合は、Software AG Installer によってサーバインスタンスが作成され、ユーザが製品選択ツリーで選択したすべての Wm パッケージおよび言語パックがパッケージリポジトリにコピーされています。インストーラで、インスタンスに選択した Wm パッケージをインストールするかどうかを選択できます(インストーラは、常に選択した言語パックをインスタンスにインストールします)。

移行できる設定、データおよびアセット

- サーバインスタンス。
- カスタム (ユーザ作成) パッケージ。
- パスワードストアとパスワード。
- Integration Server 設定ファイル。
- Trading Networks など、ホストされる製品の設定。
- JAAS 設定ファイル。
- JDBC 接続プールの設定、データベースドライバ、および機能エイリアス。
- 古い Integration Server で Wm パッケージがホストされていて、その Wm パッケージが新しい Integration Server にも存在する場合は、そのパッケージ用の設定ファイルが移行されます。
- #include ディレクティブとコメント (関連付けられたプロパティは除く) を含む、古い custom_wrapper.conf ファイルに加えられた Java サービスラッパーのカスタマイズ内容。

- 9.5 以降のカスタム (ユーザ作成) jar ファイル

移行の一部として実行されるその他のアクション

- 新しい Integration Server で使用されない古い設定ファイルを削除し、新しい Integration Server で使用されない古いプロパティを新しい設定ファイルから削除します。
- 古い Integration Server で埋め込みデータベースを使用していた場合、データベーステーブルを古い Integration Server から新しい Integration Server にコピーし、必要に応じてテーブルを新しい形式に変換します。
- Integration Server データベーステーブルと設定ファイルに保存されているサーバ ID を更新し、新しいホストマシンを反映します。
- 「Validate schemas using Xerces」という名前の新しいプロパティを既存の Web サービス記述子に追加し、この新しいプロパティに対して古い Integration Server における watt.server.wsdl.validateWSDLSchemaUsingXerces パラメータの値を設定します。新しいプロパティは、そのプロパティによって提供される機能を置き換えます。新しいプロパティについては、『*webMethods Service Development Help*』を参照してください。
- ビジネスルールデータを移行する場合は、ビジネスルールプロジェクトをアップグレードします。Java のバグにより、XML 解析メッセージが表示されることがありますが、それらのメッセージは無視してください。
- CloudStreams をアップグレードする場合は、CloudStreams サーバの管理に関連する設定アーティファクトを移行します。

Integration Server の移行

1. Integration Server クラスタをアップグレードする場合、23 ページの「Integration Server または Microservices Runtime の要件」を参照してください。
2. Deployer をホストする Integration Server をアップグレードする場合、91 ページの「Asset Build Environment、Deployer およびアプリケーションプラットフォームの移行」の手順を参照してください。
3. 新旧の Integration Server が同じマシン上にある場合、古い Integration Server がシャットダウンされていることを確認してください。
4. 移行では、古いインストールをスキャンして、カスタムパッケージの有無が確認されます。ただし、名前が Wm で始まるカスタムパッケージは、Software AG で提供されるパッケージに使用される命名規則なので、見つけられません。名前が Wm で始まるカスタムパッケージを使用している場合は、*new_Software AG_directory* /IntegrationServer/bin/migrate ディレクトリに移動して、packages.cnf ファイルを開き、カスタムパッケージを示す <value name></value> タグを追加します (例: <value name="WmFINMessages">WmFINMessages</value>)。

   メモ: 将来のアップグレードを簡素化するために、一般的なベストプラクティスとして、カスタムパッケージの名前 に Wm を付けるような命名規則を使用しないでください。

5. 次のいずれかの方法で移行できます。

   - 移行ユーティリティを使用して移行する場合は、123 ページの「移行ユーティリティを使用した製品の移行」の手順に従ってください。
   - Command Central を使用して移行する場合は、以下の Command Central コマンドを使用してください。コマンドで使用する *product_ID* を検索するには、Command Central コマンドの `sagcc list inventory products node_alias` を実行します。カスタム移行を使用する場合は、最初に移行ユーティリティを手動で実行して migrate.dat ファイルを作成し (手順については 123 ページの「移行ユーティリティを使用した製品の移行」 を参照)、`importFile` パラメータで migrate.dat ファイルを指定します。

     ```
     sagcc exec administration product target_node_alias
     integrationServer migration migrate
     {srcDir|srcFile}=full_path_to_{old_Software AG_directory|ZIP file}
     [importFile=full_path_to_migrate.dat ]
     instanceName=name [newInstanceName=name ]
     [cloneDbURL=URL cloneDbUser=user cloneDbPassword=password ]
     ```

Integration Server のアップグレードの完了

Java サービスラッパー #include ディレクティブの確認

新しい custom_wrapper.conf ファイルの最後に、すべての Java サービスラッパー #include ディレクティブが移行されます。それらを確認し、必要に応じて調整します。custom_wrapper.conf ファイルは *Software AG_directory* /profiles/IS_*instance_name* /configuration ディレクトリにあります。

メモ: 『*Software AG Infrastructure Administrator's Guide*』に記載されているように、Software AG の指示がない限り、wrapper.conf ファイルを変更しないでください。wrapper.conf ファイルを変更した場合、古い wrapper.conf ファイルで変更したプロパティの値を、新しい custom_wrapper.conf ファイルの対応するプロパティに手動でコピーします。

Integration Server 起動ファイルのカスタマイズ

Integration Server 起動スクリプトは、リリース 9.7 で変更されました。古いスクリプトをカスタマイズしていた場合は、『*webMethods Integration Server Administrator's Guide*』および Integration Server の readme を参照し、それらのカスタマイズを新しいインストール用に複製する方法を確認してください。

WSDL の更新

入力署名、出力署名、ヘッダーまたは障害メッセージで、フィールド名が `xml` で始まる動作があるプロバイダ Web サービスを使用する場合は、以下の手順に従います。

1. 新しい Integration Server を起動します。

2. *new_Software AG_directory* /install/logs ディレクトリで migrationLog.txt ファイルの内容を確認します。次のエラーが表示されている場合:

   A property watt.server.xml.ncname.encode.backward.compatibility exists in config/server.cnf with value as true.Make sure you make the required changes as specified in the upgrade documentation.Not doing so could have adverse effects as support for this property may be dropped in a future release.

   WSDL を次のように更新します。

   a. 新しい Integration Server Administrator を開き、新しい Integration Server をポイントします。

   b. [**設定**] > [拡張設定] ページに移動します。拡張設定の watt.server.xml.ncname.encode.backward.compatibility があり、この設定が True になっている場合は、False に変更します。

   c. 入力署名、出力署名、ヘッダーまたは障害メッセージで、フィールド名が `xml` で始まる動作があるすべてのプロバイダ Web サービスに対して、クライアントを再生成します。

9.7 または 9.8 からのアップグレード: シリアル webMethods メッセージングトリガーの同期および更新

新しい Integration Server で Universal Messaging を使用している場合は、順次処理をする各 webMethods メッセージングトリガーに、Universal Messaging の対応する優先名オブジェクトが必要です。ただし、古い Integration Server から移行した逐次実行トリガーは、Universal Messaging の共有された命名オブジェクトに対応します。以下のいずれかを実行して、対応する優先名オブジェクトを作成します。

- Universal Messaging が新規インストールの場合 (以前のリリースからの移行でない場合) は、Software AG Designer または組み込みサービス pub.publish:syncToProvider を使用するプロバイダを使ってパブリッシュ可能なドキュメントタイプを同期します。
- Universal Messaging が以前のリリースから移行され、かつ Integration Server は Universal Messaging に接続する Integration Server のみで、さらに Universal Messaging 接続エイリアスの共有クライアントプリフィックスプロパティが「No」に設定されている場合は、トリガーが使用する Universal Messaging 設定エイリアスを無効にしてから有効にします。
- Universal Messaging が以前のリリースから移行され、かつ Integration Server が Universal Messaging に接続する Integration Server のみでないか、共有クライアントプリフィックスが「Yes」に設定されている場合は、命名オブジェクトが完全になくなり、送信される新しいドキュメントがなくなるのを待ってから、Universal Messaging Enterprise Manager を使用して命名オブジェクトを削除します。命名オブジェクトを削除するために、ドキュメントパブリッシャーの停止が必要になる場合があります。次に、トリガーが使用する Universal Messaging 接続エイリアスを無効にしてから有効にします。

10.1 からのアップグレード: 更新されたパブリッシュ可能なドキュメントタイプの保存および同期

ソリューションに、プロトコルバッファのエンコーディング形式を使ったパブリッシュ可能なドキュメントタイプが含まれている場合、サブスクライブトリガーのキューは空になっていますが、ドキュメントタイプを Universal Messaging に編集、保存、および同期する必要があります。

Integration Server およびホストされる製品のデータベースコンポーネントへの接続

機能エイリアス、データベースドライバ、および JDBC 接続プールを移行していない場合は、Integration Server およびそれによってホストされる製品 (Trading Networks など) をデータベースコンポーネントへ接続します。手順については、『*Installing Software AG Products*』を参照してください。

ActiveTransfer のアップグレードの完了

古い ActiveTransfer インストールでカスタムのキーストアファイルを使用した場合は、新しいインストールで指定した場所で使用およびアクセスが可能であることを確認します。

アーカイブファイル共有があり、古いインストールで mft.sharing.account.tempdir プロパティを設定していなかった場合は、*new_Software AG_directory* /IntegrationServer/instances/*instance_name* /packages/WmMFT/config ディレクトリに移動し、properties.cnf ファイルを開いて、古い TempAccounts パスを mft.sharing.account.tempdir プロパティで指定します。

ActiveTransfer の起動時に、Command Central 標準に準拠するようリスナー (ポート) 名が変更されます。ActiveTransfer が重複したリスナー (ポート) 名を検出した場合、各ポート名に下線と一意の番号が追加されます。リスナー (ポート) 名に空白文字が含まれる場合、空白文字は下線に置き換えられます。

CloudStreams のアップグレードの完了

CloudStreams アーティファクトを含むカスタムパッケージをすべてバックアップしてから、WmCloudStreams パッケージ内にある `pub.cloudstreams.migration:migrate` パブリックサービスを実行します。このサービスは、WmCloudStreams パッケージに依存するカスタムパッケージ内の古い CloudStreams アーティファクトを移行し、新しい CloudStreams との互換性を持つよう更新します。アーティファクトには、SOAP や REST のコネクタサービスおよび接続が含まれます。このユーティリティは、移行の結果を Integration Server サーバログに記録します。

以下の表は、`pub.cloudstreams.migration:migrate` サービスの入力署名を示しています。

パラメータ	説明
allPackages	**Boolean string**。これはオプションです。次のように設定されます。 ■ WmCloudStreams に依存するすべてのカスタムパッケージから CloudStreams アーティファクトを移行する場合は `true`。

パラメータ	説明
	■ *packages* パラメータに指定されたカスタムパッケージからのみ CloudStreams アーティファクトを移行する場合は `false`。これがデフォルトです。
packages	**Object**。移行するカスタムパッケージの名前を含むストリング配列。少なくとも 1 つのパッケージ名を指定する必要があります。

サービスの出力署名は、iData レコードの配列で構成された Result というパラメータです。この配列には、入力署名で指定された各カスタムパッケージのレコードが含まれます。

以下の表は、`pub.cloudstreams.migration:migrate` サービスの入力署名内で指定されている各カスタムパッケージの配列内の各レコードのフィールドを一覧しています。

フィールド	説明
packageName	**String**。カスタムパッケージの名前。
success	**String**。移行が成功したか (`true`)、失敗したか (`false`) を示す値。
message	**String**。パッケージ内で見つかった CloudStreams アーティファクトの数、正常に移行された CloudStreams アーティファクトの数など、移行の情報。
info	**Object**。これはオプションです。移行中にサービスが info メッセージを発行した場合におけるメッセージのストリング配列。
errors	**Object**。これはオプションです。移行中にサービスがエラーを発行した場合におけるエラーのストリング配列。

TLS 1.1 および 1.2 トラストストアの設定

JVM トラストストアの一部ではない認証を持つ SaaS バックエンドに対し、SaaS バックエンドの認証を使用して Integration Server に新しい JKS トラストストアを作成および適用します。トラストストアは認証チェーン内にすべての認証を含んでいる必要があります。これはセキュアな認証交換を設定するための一般的なプラクティスです。このような場合、トラストストアがなければ、認証に関するエラーが発生します。

1. Integration Server Administrator で、[**セキュリティ**] > [キーストア] ページに移動して、トラストストアを作成します。詳細な手順については、『*webMethods Integration Server Administrator's Guide*』の通信のセキュリティに関するセクションを参照してください。

   メモ: 公開されている入手可能なツールを使用してトラストストアを作成することもできます。

2. 接続の拡張プロパティ [**トラストストアエイリアス**] でトラストストアを適用します。詳細な手順については、使用中のプロバイダに対する CloudStreams マニュアルを参照してください。

Dynamic Business Orchestrator のアップグレードの完了

old_Software AG_directory /IntegrationServer/instances/*instance_name* /packages/WmDBO/config ディレクトリに移動して、provision.cfg ファイルを新しいインストールディレクトリの同じ場所にコピーします。

アダプタのアップグレードの完了

古い Integration Server でアダプタがホストされており、新しい Integration Server で同じリリース以降のアダプタをインストールした場合は、Integration Server 移行の一部としてそれらのアダプタのデータが移行されます。ただし、以下のアダプタについては、指示されたタスクを完了する必要があります。Integration Server を再起動します。

ホストの Integration Server をアップグレードせずにアダプタをアップグレードする手順については、アダプタ製品のマニュアルを参照してください。

webMethods Adapter for JDBC

アダプタが OCI ドライバを使用して Oracle RDBMS に接続する場合は、新しい Integration Server マシンで環境変数を設定します。

以下の表は、Integration Server マシン用のリストされたプラットフォームに対する環境変数設定を示しています。

プラットフォーム	環境変数の設定
Solaris	LD_LIBRARY_PATH=¥ORACLE_HOME¥lib
HP	SHLIB_PATH=¥ORACLE_HOME¥lib
AIX	LIBPATH=¥ORACLE_HOME¥lib
Linux	LD_LIBRARY_PATH=¥ORACLE_HOME¥lib

webMethods Microsoft Package

古い Integration Server と同じホスト上にある新しい Integration Server に webMethods Microsoft Package を移行する方法については、製品マニュアルを参照してください。

webMethods Adapter for SAP

1. 新しい Integration Server で UNIX システムを使用している場合は、そのマシンで環境変数を設定します。

   以下の表は、Integration Server マシン用のリストされたプラットフォームに対する環境変数設定を示しています。

プラットフォーム	環境変数の設定
Linux、Solaris	LD_LIBRARY_PATH
HP	SHLIB_PATH
AIX	LIBPATH

2. 新しい Integration Server で Windows システムを使用している場合は、古い Integration Server マシン上の C:¥WINDOWS¥system32¥drivers¥etc ディレクトリに移動し、services ファイルを開き、「sapgw」という語句が含まれているすべてのエントリを新しいインストールの同じファイルにコピーします。たとえば、sapgw00 3300/tcp、sapgw01 3301/tcp、sapgw02 3302/tcp、sapgw03 3303/tcp、sapgw04 3304/tcp、sapgw05 3305/tcp などのエントリがあります。

eStandards モジュールのアップグレードの完了

古い Integration Server で eStandards モジュールがホストされていて、新しい Integration Server で同じリリース以降のモジュールをインストールした場合は、Integration Server 移行の一部としてそれらのモジュールのデータが移行されます。ただし、以下のモジュールについては、指示されたタスクを完了する必要があります。Integration Server を再起動します。

ホストの Integration Server をアップグレードせずに eStandards モジュールをアップグレードする手順については、eStandards モジュール製品ドキュメントを参照してください。

webMethods SWIFTNet Module

モジュールを再設定します。手順については、『*webMethods SWIFTNet Module Installation and User's Guide*』を参照してください。

12 Software AG Designer、ビジネスプロセスランタイムデータ、および Closed Loop Analytics の移行

- Software AG Designer の移行 78
- ビジネスプロセスランタイムデータの移行 82
- Closed Loop Analytics JMS トピックの移行 83

Software AG Designer の移行

移行の前に

1. 新しい Software AG Designer を開き、新しいワークスペースをポイントします。たとえば、デフォルトの workspace*release* をそのまま使用できます (workspace912 など)。
2. 必要に応じて、サードパーティ機能 (たとえば Subversion のサポート) をインストールします。新しい Software AG Designer にインストールされている Eclipse のリリースは Eclipse 4.7.3 であるため、追加する機能がこの Eclipse のリリースと互換性があることを確認します。
3. アップグレード前に設定をエクスポートしていた場合は、以下のようにインポートします。
   a. [**File**] > [Import] ウィザードを選択します。[Select] パネルで、[**General**] > [Preferences] を選択して、[**Next**] をクリックします。[**From preference file**] フィールドに、設定をエクスポートした .epf ファイルを指定します。[**Finish**] をクリックします。
   b. [**Window**] > [Preferences] を選択します。[Preferences] ウィンドウで [**Java**] > [Installed JREs] > [Execution Environments] ページに移動します。複数の JRE がリストされている場合は、JRE 1.8 がデフォルトとして選択されていることを確認するか、またはリストから古い JRE を削除します。古い製品インストールをポイントしている設定を更新して、新しい製品インストールをポイントするように設定します。たとえば My webMethods Server の場合、[**Server**] > [Runtime Environments] 設定を更新します。Application Platform では、ランタイムインスタンスへのパスを更新し、次に [Error] タブでメッセージを確認し、更新する必要があるパスがあるか確認します。
   c. Software AG Designer を再起動します。

Apama、Application Platform、Business Process および Business Rule プロジェクトの移行

1. Apama を移行する場合、変更したファイルを古い作業ディレクトリから新しい作業ディレクトリに移行します。
2. プロジェクトがソースコントロールシステムに保存されている場合は、そのシステム用のインポートウィザードを使用します (たとえば [**Import SVN**] > [Project from SVN])。

   プロジェクトが Software AG Designer ワークスペースに保存されている場合は、以下の手順に従います。

   a. [**File**] > [Import] ウィザードを選択します。
   b. [Select] パネルで、[**General**] > [Existing Projects into Workspace] を選択して [**Next**] をクリックします。
   c. [Import Projects] パネルで、以下の手順に従います。

a. [**Select Root Directory**] をクリックして、インポートする種類のプロジェクトがあるワークスペースに移動します。または、[**Select Archive File**] をクリックして、インポートする種類のプロジェクトがあるディレクトリに移動します。

b. [**Projects**] ボックスで、インポートするプロジェクトを選択します。[**Copy projects into workspace**] を選択します。[**Finish**] をクリックします。

3. Apama を移行する場合、[**Project**] > [**Clean**] を使用して、Software AG Designer の制御下にある生成されたプロジェクトのアーティファクトを強制的に再構築します。

4. プロジェクトを確認します。

以下の表は、一覧された製品のプロジェクトを確認する方法を説明しています。

製品	確認
Apama	すべてのプロジェクトの構築でエラーが出ない。[**Problems**] タブにエラーが表示された場合は、各エラーをクリックして解決します。
Application Platform	すべてのプロジェクトのコンパイルでエラーが出ない。[**Problems**] タブにエラーが表示された場合は、各エラーをクリックして解決します。たとえば、クラスをタイプに解決できないことを示すエラーが表示された場合は、プロジェクトのクラスパスの設定を編集します。
Business Process	インポートしたすべてのビジネスプロセスプロジェクトが [**ソリューション**] タブの [プロセス] ノードの下に表示されている。
Business Rules	インポートしたすべてのビジネスルールプロジェクトが [**ソリューション**] タブの [ルール] ノードの下に表示されている。

5. ビジネスルールプロジェクトをインポートした場合は、[ルールエクスプローラ] で各プロジェクトを右クリックし、[**プロジェクトのアップグレード**] をクリックします。

6. Apama を移行する場合、次の操作を実行します。

a. プロジェクトで Digital Event Services からイベントタイプをインポートする場合、それらのイベントタイプの再インポートが必要になることがあります。プロジェクトを開き、config/connectivity/DigitalEventServices の下にある EventTypeList ファイルを開きます。次に、[Sync from Digital Event Services] をクリックします。変更がある場合、ファイルを保存します。

b. ant を使用してプロジェクトの構築と展開を自動化する場合は、ant 展開スクリプトを空のディレクトリにエクスポートします。スクリプトを実行して、プロジェクト内の生成されたアーティファクトの再構築を強制的に実行します (Apama クエリー、シナリオ、Apama API を使用した Java ¥C¥C++¥.NET コードなど)。エラーや警告については、構築の出力を確認します。

c. Apama ダッシュボード展開を使用する場合は、新しい展開パッケージを作成し、展開パッケージをインストールします。

タスクアプリケーションプロジェクトの移行

1. 設定を更新します。
   a. [Window] > [Preferences] > [Server] > [Runtime Environments] を選択します。インストールされているサーバランタイムのリストにこのリリースの My webMethods Server が含まれていない場合は、このランタイムを追加します。
   b. [**Software AG**] > [タスク開発] に移動します。設定ウィンドウで、その他のタスク関連の Software AG Designer 設定がすべて正しいことを確認し、[**OK**] をクリックします。
2. プロジェクトをインポートします。
   a. [**File**] > [Import] ウィザードを選択します。
   b. [Select] パネルで、[**Software AG**] > [既存の CAF プロジェクトをワークスペースへ] に移動して [**Next**] をクリックします。
   c. [Import Projects] パネルで、[**Select Root Directory**] をクリックして、タスクアプリケーションプロジェクトがあるワークスペースまたはソースコントロールシステムに移動します。または、[**Select Archive File**] をクリックして、タスクアプリケーションプロジェクトがあるディレクトリに移動します。[**Projects**] ボックスで、インポートするプロジェクトを選択します。
   d. [**Copy projects into workspace**] を選択し、[**Finish**] をクリックします。
3. インポートしたすべてのタスクのアプリケーションプロジェクトが [**ソリューション**] タブの [タスク] ノードの下に表示されていることを確認してください。
4. エラーが [**Problems**] タブに表示される場合、[**Navigator**] タブをクリックします。各タスクアプリケーションプロジェクトを右クリックして、[**CAF ツール**] > [**CAF プロジェクトの修復**] をクリックします。エラーが消えない場合は、Software AG Designerを再起動してください。
5. 移行されたタスクアプリケーションプロジェクトを、新しいリリースから My webMethods Server に公開します。手順については、『*webMethods BPM Task Development Help*』を参照してください。

CAF アプリケーションの添付リストコントロールのカスタマイズのやり直し

9.10 では、CAF **添付リスト**制御の機能が強化されています。たとえば、Java アプレットベースのドラッグアンドドロップパネルは、HTML5 ベースに変わりました。

My webMethods Server 移行ユーティリティを実行したときに、CAF アプリケーションは移行されますが、CAF アプリケーション内の**添付リスト**コントロールへのカスタマイズは移行されません。移行した CAF アプリケーションのカスタマイズを手動でやり直します。

CloudStreams のサーバ定義とプロジェクトの移行

1. [Window] > [Preferences] > [Software AG] > [CloudStreams] > [サーバ] に移動します。[インポート] をクリックして、エクスポートした .properties ファイルを選択し、[開く] をクリックしま

す。Software AG Designer は、インポートする定義で、既存のサーバ定義を上書きするかどうかを尋ねます。[OK] を 2 回クリックします。

2. [CloudStreams 開発] パースペクティブを開きます。
3. [CloudStreams Governance] ビューで空白を右クリックし、[Import] をクリックして、ウィザードを開きます。[ルートディレクトリの選択] フィールドで、エクスポートした Governance プロジェクトを含むディレクトリを指定します。[プロジェクト] ボックスで、エクスポートしたプロジェクトのフォルダからインポートする Governance プロジェクトを選択します。[プロジェクトをワークスペースにコピー] を選択して、[Finish] をクリックします。[Governance] タブをクリックして、インポートしたすべての Governance プロジェクトが表示されていることを確認します。
4. [CloudStreams プロバイダ] ビューで、空白を右クリックします。アーカイブ (圧縮ファイル) にエクスポートしたプロバイダプロジェクトをインポートする場合は、[アーカイブのインポート] をクリックします。別の CloudStreams からプロバイダプロジェクトをインポートする場合は、[ワークスペースからインポート] をクリックします。

Integration Server または Microservices Runtime 定義の移行

1. [**Window**] > [Preferences] を選択します。[**Preferences**] ダイアログボックスにある左側のナビゲーションバーで、[**Software AG**] > [Integration Server] を選択します。
2. [**インポート**] をクリックして、エクスポートした .properties ファイルを選択し、[**開く**] をクリックします。Software AG Designer によって既存のサーバを上書きするかどうかを尋ねられます。[**OK**] をクリックし、もう一度 [**OK**] をクリックして [**Preferences**] ダイアログボックスを閉じます。

モバイルプロジェクトの移行

1. [Window] > [Preferences] > [Software AG] > [モバイル開発] に移動し、Mobile Designer インストールディレクトリを、新しい Software AG Designer インストールディレクトリ内の **MobileDesigner サブディレクトリに変更します。**
2. プロジェクトのインポート
   a. [**File**] > [Import] ウィザードを選択します。
   b. [Select] パネルで、[General] > [Existing Projects into Workspace] を選択して [Next] をクリックします。
   c. [Import Projects] パネルで、[Select Root Directory] をクリックし、使用するモバイルプロジェクトが含まれているワークスペースに移動します。[**Projects**] ボックスで、インポートするプロジェクトを選択します。
   d. [**Copy projects into workspace**] を選択し、[**Finish**] をクリックします。
3. [Package Explorer] で、以下の手順に従います。問題やエラーを解決する手順については、『*webMethods Mobile Development Help*』の「*Migrating Mobile Projects*」を参照してください。
   a. 各インポートプロジェクトのモデルディレクトリに移動し、.aml ファイルをダブルクリックします。警告とエラーを確認して、問題を解決します。

b. インポートされた各プロジェクトのルートノードを右クリックして、[ソースコードの生成] > [アプリケーションモデルと API] をクリックします。コンパイルエラーを解決します。

c. (Jenkins-based Remote Multi-Build の代わりに) Mobile Administrator を使用している場合、Mobile Administrator Remote-Multi-Build ant ターゲットを有効にします。プロジェクトのルートに移動して、build.xml ファイルを開いて、以下の文字列をファイルの最後に 1 行で追加します。

```
<import file="${env.MOBILE_DESIGNER}/plugins/MobileAdministrator/
v1.0.0/targets.xml"/>
```

4. 後から、Android ハンドセットターゲットで、Phoney シミュレータ内でプロジェクトを実行しているときの問題のような問題が発生した場合、[Package Explorer] で各プロジェクトの /target ディレクトリに移動し、すべての既存のハンドセットターゲットを削除します。そして、++Activate-Handset ant ターゲットを実行することにより、プロジェクトのターゲットを再追加します。

ビジネスプロセスランタイムデータの移行

メモ: アップグレードする前に作成したプロセスモデルに基づくプロセスインスタンスが稼働中で、新しい Software AG Designer　でそのプロセスモデルを再生成する場合は、再生成する前にプロセスモデルのバージョン番号を次のシーケンス番号に変更します。そうしないと、プロセスインスタンスが想定通りに動作しません。My webMethods Server で実行するためにプロセスモデルの新しいバージョンを有効にすると、稼働中のプロセスをアップグレードするかどうかの確認を求められますが、[いいえ] と答えます。

メモ: Microservices Runtime の手順は Integration Server の場合と同じです。

1. Software AG Designer でユーザが作成したビジネスプロセスパッケージを移行した場合、パッケージが *new_Software AG_directory* /IntegrationServer/instances/*instance_name* /packages ディレクトリに存在していることを確認します。パッケージ名は、プロジェクト名または Software AG Designer で指定したカスタムの名前です。

2. Integration Server Administrator を開き、プロセスエンジンをホストする新しい Integration Server をポイントします。[**設定**] > [JDBC プール] ページに移動します。ProcessEngine および ProcessAudit 機能をデータベースコンポーネントに接続します。手順については、『*Installing Software AG Products*』を参照してください。

3. Integration Server Administrator では、[**パッケージ**] > [管理] ページに移動して、WmMonitor パッケージの アイコンをクリックします。

4. 新旧の製品インストールが異なるマシン上にあり、アップグレード前に実行したプロセスとプロセスステップを再サブミットする場合は、以下の手順に従います。

   Monitor をホストする新しい Integration Server がメッセージング製品に接続されており、プロセスモデルフラグメントがある場合、[ローカル IS に再サブミット] チェックボックスをオンにして、[サブミット] をクリックし、パッケージを再ロードします。

   Monitor をホストする新しい Integration Server がメッセージング製品に接続されておらず、プロセスモデルフラグメントもない場合は、[ローカル IS に再サブミット] チェックボックスをオフにして、[サブミット] をクリックし、パッケージを再ロードします。

5. [設定] > [メッセージング] > [webMethods メッセージングトリガーの管理] ページに移動して、新しい Integration Server 上のすべての webMethods メッセージングトリガーに対してドキュメント抽出が有効になっていることを確認します。手順については、『*webMethods Integration Server Administrator's Guide*』を参照してください。

Closed Loop Analytics JMS トピックの移行

78 ページの「Software AG Designer の移行」で、Closed Loop Analytics に必要となるすべての設定とデータを移行しました。ここで、Closed Loop Analytics がエスカレーションイベントを収集する、Universal Messaging の JMS トピックを更新する必要があります。

1. BPM 用に使用している新しい Universal Messaging のインスタンスを起動してから、Enterprise Manager を起動します。
   - 左のパネルで領域ツリーを開き、BPM 用に使用している Universal Messaging インスタンスをクリックします。
   - 右のパネルで [JNDI] タブをクリックして、トピックツリーを開きます。エスカレーションイベントタイプのトピック {http://namespaces.softwareag.com/EDA/WebM/Rules/CLA/1.0}ProcessEscalation を削除します。
2. Universal Messaging インスタンスのホストマシンで、*new_Software AG_directory* /jvm/jvm/bin ディレクトリに移動して、コマンドウィンドウまたはシェルを開きます。以下のコマンドを入力して、エスカレーションイベントタイプの新しいトピックを作成します。以下で表示されている改行部分をスペースにし、1 行のコマンドとして入力します。

   ```
   java -jar new_Software AG_directory  /common/lib/event-routing-jms-util.jar
   bind nsp://locahost:9000
   {http://namespaces.softwareag.com/EDA/WebM/Rules/CLA/1.0}ProcessEscalation
   ```

13 Optimize および Infrastructure Data Collector の移行

- Terracotta Server Array の設定と起動 86
- 他の製品への接続の更新 86
- Infrastructure Data Collector の移行 87
- Optimize 環境の展開 88
- 9.7 からのアップグレード: 4.1 以降の Terracotta Server Array の監視の開始 90
- webMethods Broker から Universal Messaging への切り替え 90

Terracotta Server Array の設定と起動

新リリースで Analytic Engine をクラスタ化している場合は、Terracotta Server Array を設定して起動する必要があります。手順については、『 *Using Terracotta with webMethods Products*』、『*Configuring BAM*』および Terracotta BigMemory Max のマニュアルを参照してください。

他の製品への接続の更新

Process Engine への接続の更新

Optimize for Process を使用している場合は、以下の手順に従います。

1. Integration Server Administrator を開き、Process Engine をホストする新しい Integration Server をポイントします。
2. [**パッケージ**] > [管理] ページに移動して、WmPRT パッケージの アイコンをクリックします。
3. 左側のナビゲーションバーで [**設定**] をクリックして、[**Process Engine 設定の編集**] をクリックします。
4. [JMS サーバ URL] フィールドで、 Universal Messaging ホスト名とポートを nsp://*host* :*port* と指定します。デフォルトポートは 9000 です。
5. [**サブミット**] をクリックして、WmPRT パッケージを再ロードします。
6. Integration Server をホスティングするすべての新しい Process Engine について、上記の手順を繰り返します。

Optimize Support パッケージへの接続の更新

Optimize 組み込みサービスを使用している場合は、以下の手順に従います。

メモ: Microservices Runtime の手順は Integration Server の場合と同じです。

1. Integration Server Administrator を開き、Optimize Support パッケージをホストする新しい Integration Server をポイントします。
2. [**パッケージ**] > [管理] ページに移動して、WmOptimize パッケージの アイコンをクリックします。
3. Analytic Engine のホストマシンとポートを指定します。デフォルトポートは 12503 です。
4. [JMS サーバ URL] フィールドで、 Universal Messaging ホスト名とポートを nsp://*host* :*port* と指定します。デフォルトポートは 9000 です。
5. [**サブミット**] をクリックして、WmOptimize パッケージを再ロードします。
6. Optimize Support パッケージをホスティングするすべての Integration Server について、上記の手順を繰り返します。

Infrastructure Data Collector の移行

移行できる Infrastructure Data Collector 設定、データおよびアセット

- アセット設定ファイル。
- SNMP、JAAS、JMX、SSH、セキュリティ、ログ、プラットフォームデバッグ、Web Services Stack、およびウォッチドッグの設定。
- #include ディレクティブとコメント (関連付けられたプロパティは除く) を含む、古い custom_wrapper.conf ファイルに加えられた Java サービスラッパーのカスタマイズ内容。

Infrastructure Data Collector の移行

1. 新旧の Infrastructure Data Collector が同じマシン上にある場合、古い Infrastructure Data Collector がシャットダウンされていることを確認してください。
2. SNMP アセット設定ファイルを移行する場合は、*new_Software_AG_directory* infrastructuredc/bin/migrate/*old_release* ディレクトリに移動して、snmpMigration.properties ファイルを開きます。古い SNMP 設定ディレクトリと新しい SNMP 設定ディレクトリへの完全パスを入力します。ファイルのコンテンツに使用する形式が表示されます。
3. 次のいずれかの方法で移行できます。
   - 移行ユーティリティを使用して移行する場合は、123 ページの「移行ユーティリティを使用した製品の移行」の手順に従ってください。
   - Command Central を使用して移行する場合は、以下の Command Central コマンドを使用してください。カスタム移行を使用する場合は、最初に移行ユーティリティを手動で実行してカスタム migrate.dat ファイルを作成し (手順については123 ページの「移行ユーティリティを使用した製品の移行」 を参照)、`importFile` パラメータで migrate.dat ファイルを指定します。

     ```
     sagcc exec administration product target_node_alias
     InfrastructureDC migration migrate
     {srcDir|srcFile}=full_path_to_{old_Software AG_directory|ZIP file}
     [importFile=full_path_to_migrate.dat ]
     ```

Infrastructure Data Collector のアップグレードの完了

新しい custom_wrapper.conf ファイルの最後に #include ディレクティブが移行されます。それらを確認し、必要に応じて調整します。custom_wrapper.conf ファイルは *Software AG_directory* /profiles/InfraDC/configuration ディレクトリにあります。

メモ: 『*Software AG Infrastructure Administrator's Guide*』に記載されているように、Software AG の指示がない限り、wrapper.conf ファイルを変更しないでください。ただし、古い wrapper.conf ファ

イルを間違えて変更した場合は、古い wrapper.conf ファイル内で変更したプロパティの値を、新しい custom_wrapper.conf ファイルの対応するプロパティに手動でコピーします。

Optimize 環境の展開

特に記述がない限り、これ以降の節で説明する Optimize の手順の詳細については、『*Configuring BAM*』を参照してください。

古い Optimize CCS 環境定義を再利用しない場合

古い CCS (Central Configurator System) 環境定義を再利用しない場合は、新しい My webMethods Server を起動して My webMethods を開き、新しい Optimize 環境を設定します。

Infrastructure Data Collector のホストを定義する場合、[**ホストの追加/編集**] ダイアログボックスの [ホスト名または IP アドレス] フィールドに、古いリリースと同じ値を入力します。

メモ: 新しい値が古い値と一致しない場合、アップグレードの完了後に、既存のアセットの監視を停止してこれらを再検出し、監視対象として選択する必要があります。

新しい Optimize 環境を展開する以下の手順に進みます。

古い Optimize CCS 環境定義を再利用する場合

新しい Optimize をインストールする前に、ファイルにエクスポートした古い CCS 環境定義を再利用する場合は、以下の手順に従います。

1. 新しい My webMethods Server を起動し、My webMethods を開きます。
2. 以下の手順に従って、古い CCS 環境をインポートします。
   a. [**アプリケーション**] > [管理] > [システム全体] > [環境] > [環境の定義] ページに移動します。
   b. [**環境のインポート**] をクリックします。[環境のインポート] ダイアログボックスで [**参照**] をクリックし、古い CCS 環境をエクスポートしたファイルを選択して [**OK**] をクリックします。プロンプトが表示されたら、移行を確認します。
   c. 「環境を最新バージョンに移行しますか?」と尋ねられたら、[**はい**] をクリックします。CCS は古い環境をインポートして、そのデータを新しい Optimize 論理サーバに移行します。
3. インポートした環境に関連付けられたデータベースプール定義に対応するように、以下の手順に従って、データベースプール接続設定を更新します。
   a. [**アプリケーション**] > [管理] > [システム全体] > [環境] > [データベースプール設定] ページに移動します。
   b. [**プール**] 列で、インポートしたデータベースプールをクリックします。
   c. [**プールの設定**] 領域で、[**接続数の最小値**] フィールドに 8 を指定し、[**接続数の最大値**] フィールドに 60 を指定します。

d. データベース接続情報が正しいことを確認します。[**データベース接続**] 領域で [**テスト**] をクリックし、ページの下部にメッセージ [テストに合格しました。] が表示されることを確認します。

e. [**保存**] をクリックします。

4. [**アプリケーション**] > [管理] > [システム全体] > [環境] > [環境の定義] ページに移動します。[**環境名**] の下で、インポートした環境をクリックします。

5. 新しいリリースで、古いリリースで使用していたものとは異なる JMS プロバイダを使用している場合は、以下の手順に従います。

a. [**サーバの設定**] タブをクリックします。[**設定**] で、[**デフォルトの設定**] ツリーを開いて [**JNDI Configuration**] をクリックします。

b. [**Naming Factory Type**] リストで、新しいリリース (Universal Messaging) で使用している JMS プロバイダをクリックし、[**Broker Name**] フィールドを消去します。[**保存**] をクリックします。

c. [**エンドポイントのマップ**] タブをクリックします。JMS プロバイダの最初の [受信接続] フィールドで、Universal Messaging 用に `nsp` と入力します。2 番目のフィールドに、ポート番号を入力します。

d. [**保存**] をクリックします。

6. 環境の妥当性を検査するには、以下の手順に従います。

a. 赤い円を表示している各タブをクリックします ([**検査**] タブを除く)。タブの設定を変更し、このタブに [**保存**] ボタンがある場合は、[**保存**] をクリックします。

b. [**検査**] タブを除くすべてのタブに緑のチェックマークが表示されたら、[**検査**] タブをクリックします。

7. [**Finish**] をクリックします。

新しい Optimize 環境の展開

1. インストールした新しい Optimize コンポーネントを起動します。これらのコンポーネントには、Analytic Engine、Infrastructure Data Collector および Web Services Data Collector などがあります。

2. 新しいリリースで使用している Universal Messaging を起動します。

3. 新しい Optimize 環境を展開します。手順については、『*Configuring BAM*』を参照してください。

4. My webMethods で、[**My webMethods**] > [システム設定] > [サーバ] ページに移動します。

- BPMS がインストールされている場合は、[**BPM および BAM**] オプションを選択し、新しい Analytic Engine、および Monitor をホストする Integration Server を指定します。ホストには DNS 名または IP アドレスを指定できます。

- BPMS がインストールされていても BPM を使用していない場合 (つまり Optimize のみを使用している場合)、[**BAM のみ**] オプションを選択し、新しい Analytic Engine を指定します。ホストには DNS 名または IP アドレスを指定できます。

5. Analytic Engine を起動してから、[**サーバ状態の確認**] をクリックしてサーバが使用可能であるか確認します (緑色のアイコン)。サーバが起動するまで少し時間がかかる場合があります (15 分程度)。次に [**保存**] をクリックします。

9.7 からのアップグレード: 4.1 以降の Terracotta Server Array の監視の開始

各手順については、『*Administering webMethods Optimize*』を参照してください。

1. 新しい Infrastructure Data Collector を起動します。
2. [**アプリケーション**] > [管理] > [インフラストラクチャコンポーネント] > [検出] に移動します。
3. すべての Terracotta (アセット) エントリを削除します。
4. 監視する 4.1 以降の Terracotta Server Array を再検出します。
5. [**アプリケーション**] > [管理] > [インフラストラクチャコンポーネント] > [監視中のコンポーネント] に移動します。Terracotta Server Array モニタ、コンポーネントおよび KPI を選択します。

webMethods Broker から Universal Messaging への切り替え

古いリリースでは JMS プロバイダとして webMethods Broker を使用していて、新しいリリースでは Universal Messaging を使用している場合、以下の手順に従います。

1. Universal Messaging を起動して、Universal Messaging Enterprise Manager を開きます。
2. Universal Messaging インスタンスに接続して、これをクリックします。
3. [**Config**] タブをクリックします。[**グローバル値**] を展開して、AllowRealmAdminFullAccess を「true」に設定します。

14 Asset Build Environment、Deployer およびアプリケーションプラットフォームの移行

- Asset Build Environment の移行 92
- Deployer の移行 92
- Application Platform の移行 92

Asset Build Environment の移行

old_Software AG_directory /common/AssetBuildEnvironment/master_build ディレクトリに移動して、build.properties ファイルを新しいインストールディレクトリの同じ場所にコピーします。

Deployer の移行

移行される設定、データおよびアセット

- 設定。
- サーバエイリアス。
- ターゲットグループ。
- プロジェクト。

メモ: Deployer プロジェクトを移行しても、それらのプロジェクトの製品アセットは古いリリースから新しいリリースに変換されません。このマニュアルの前述の手順に従ってアセットを移行する必要があります。

Deployer の移行

1. このマニュアルの前述の手順に従って、古い Deployer で定義されていたソース/ターゲットの全製品サーバを新しいリリースにアップグレードして、それらの製品のすべての設定、データ、およびアセットを新しいリリースに移行します。
2. 新旧の Deployer が同じマシン上にある場合、古い Deployer がシャットダウンされていることを確認してください。
3. ホストの Integration Server を移行する際に、Deployer を移行します。Deployer の移行前に Integration Server を起動しないでください。起動してしまった場合は、MIGRATION_DONE という名前のファイルが WmDeployer パッケージに作成され、移行を実行できなくなります。新しいホストの Integration Server を起動した場合は、このファイルを削除してください。
4. 次のいずれかの方法で移行できます。
   - 移行ユーティリティを使用して移行する場合は、Integration Server の 123 ページの「移行ユーティリティを使用した製品の移行」 の手順に従ってください。
   - Command Central を使用して移行する場合は、67 ページの「Integration Server、Microservices Runtime、およびホストされている Wm パッケージの移行」 の Integration Server の手順に従ってください。

Application Platform の移行

移行される Application Platform 設定、データ、および、アセット

- Integration Server インストール内に置かれている Application Platform 設定ファイル。
- My webMethods Server インストール内に置かれている Application Platform プロジェクトバンドル。
- 9.8、9.9、9.10 または 9.12 からのアップグレード: Application Platform Web アプリケーションとして使用する WmTomcat を使用して作成された Web アプリケーション各 Web アプリケーションは war 圧縮ファイルとして *new_Software AG_directory* /profiles/*instance_name* /workspace/webapps ディレクトリに移行されます。

Application Platform の移行

1. Application Platform Support パッケージをホストする新旧の Integration Server インスタンスが同じマシン上にある場合は、古いインスタンスがシャットダウンされていることを確認します。
2. Application Platform プロジェクトバンドルをホストする新旧の My webMethods Server インスタンスが同じマシン上にある場合、古いインスタンスがシャットダウンされていることを確認してください。
3. 次のいずれかの方法で移行できます。
   - 移行ユーティリティを使用して移行する場合は、123 ページの「移行ユーティリティを使用した製品の移行」の手順に従ってください。
   - Command Central を使用して移行する場合は、以下の Command Central コマンドを使用してください。

     ```
     sagcc exec administration product target_node_alias
     PLSCore migration migrate
     {srcDir|srcFile}=full_path_to_{old_Software AG_directory|ZIP file}
     ```

Application Platform のアップグレードの完了

パッケージのインポートの確認と修正

new_Software AG_directory /install/log ディレクトリに移動し、migrationLog.txt ファイルを開きます。このメッセージと似ているメッセージを確認します。

```
Project bundle /dev/installs/99oct2015/profiles/IS_default/
workspace/app-platform/deployer/bundles/greeter-web.jar
contains a package com.softwareag.applatform.sdk with import version
range [start_version , end_version ) that must be upgraded manually.
```

このメッセージは Application Platform プロジェクトで使用されたパッケージのセマンティックバージョニングに関係しています。詳しい情報については、OSGi Alliance の『Semantic Versioning Technical Whitepaper』を参照してください。

メッセージが存在し、Application Platform プロジェクトのマニフェストファイルを作成したことがない場合は、Asset Build Environment と Deployer を使用してプロジェクトを再構築および再展開します。

メッセージが存在し、Application Platform プロジェクトのマニフェストファイルを作成した場合は、マニフェストファイルを編集して *end_version* を含め、プロジェクトを再構築および再展開します。

Application Platform プロジェクトを再構築および再展開する手順については、『*webMethods Application Platform User's Guide*』を参照してください。

Tomcat 設定の更新

1. *old_Software AG_directory* /IntegrationServer/web/conf ディレクトリに移動して、ディレクトリ内のすべての設定ファイルをテキストエディタで開きます。
2. *new_Software AG_directory* /profiles/*instance_name* /configuration/tomcat/conf ディレクトリに移動して、ディレクトリ内のすべての設定ファイルをテキストエディタで開きます。
3. 「https://tomcat.apache.org/migration-85.html」 に移動します。
4. Web ページの手順に従って、古い設定ファイルに作成されたものと同じカスタム変更を新しい設定ファイルにも作成するように変更します。
5. *new_Software AG_directory* /profiles/*instance_name* /configuration/tomcat/conf ディレクトリに移動し、server.xml ファイルを開いて、既存の <Realm> エレメントを以下のものと置き換えます。

```
<Realm className="org.apache.catalina.realm.LockOutRealm">
  <Realm className="com.softwareag.platform.catalina.auth.SINRealm"
    name="AppPlatformRealm"
    userClassNames="com.softwareag.security.jaas.principals.SagUserPrincipal"
    roleClassNames="com.softwareag.security.jaas.principals.SagRolePrincipal"
    defaultRealm="AppPlatformRealm"/>
</Realm>
```

9.8、9.9、9.10 または 9.12 からのアップグレード: WmTomcat Web アプリケーションの移行の完了

1. WmTomcat 上の各 Web アプリケーションの依存関係を削除します。移行されたそれぞれの Integration Server パッケージに対して、*new_Software AG_directory* /IntegrationServer/instances/*instance_name* /*migrated_package* ディレクトリに移動して、テキストエディタで manifest.v3 ファイルを開き、以下の行を削除してファイルを保存します。

   ```
   <value name="WmTomcat">...</value>
   ```

2. 移行された Web アプリケーションの URL は元の Web アプリケーションの URL と異なります。ポート番号は変更され、/web セグメントは削除されます (例: 古い URL http://localhost:5555/web/MyPackage/index.jsp は http://localhost:8072/MyPackage/index.jsp になります)。ユーザに通知し、URL に依存したソフトウェアコンポーネントを更新します。
3. 各 Web アプリケーションは war 圧縮ファイルとして移行されます。Tomcat サーブレットコンテナの開始にかかる時間を減らしたい場合は、*new_Software AG_directory* /

profiles/*instance_name* /workspace/webapps ディレクトリに移動し、各圧縮ファイルから war ファイルを webapps ディレクトリへ展開します。

4. Web アプリケーションの JSP ページの中に Integration Server API com.wm.app.b2b.server.Service.doInvoke( ) への直接呼び出しがある場合は、〈webm:invoke/〉カスタムタグと置き換えてください。手順については、Application Platform のマニュアルを参照してください。

15 Apama の移行

1. 新旧の Apama が別々のマシンにある場合は、展開スクリプト (ant スクリプトとプロパティファイル) を更新して、新しいインストールの正しいホスト名を指定します。
2. 新旧の Apama が同じマシン上にある場合は、古いインストールがシャットダウンされていることを確認します。
3. ant スクリプトを実行して、プロジェクト内の生成されたアーティファクトの再構築を強制的に実行します (Apama クエリ、シナリオ、Apama API を使用した Java/C/C++/.NET コードなど)。構築の出力でのエラーと警告の確認
4. UAT (User Acceptance Testing) 環境で実行し、移行されたアプリケーションを新しい Apama に注入します。コリレータログのエラーまたは警告をスキャンします。すべてのシステムテスト (PySys テストケースなど) を再実行し、アプリケーションが予想通りに動作することを確認します。
5. Command Central を使用している場合、古いインストールにインスタンスを作成したときと同じコマンドライン、GUI、またはコンポジットテンプレートプロシージャを使用して、Apama リリースノートに記載されている廃止または削除されたアイテムに対応した変更を加えつつ、新しいインストールに Apama インスタンスを再作成します。

16 API Gateway、API Portal、および CentraSite の移行

- 10.1 からのアップグレード: API Gateway の移行 100
- API Portal の移行 100
- CentraSite の移行 102

10.1 からのアップグレード: API Gateway の移行

移行される API Gateway 設定、データおよびアセット

- Event Data Store (35 ページの「10.1 からのアップグレード：古い API Gateway の準備」 を参照)
- API Gateway、API Gateway UI および ダッシュボード (Kibana) をホストする古い Integration Server の設定、および 1 つの API Gateway インスタンス

API Gateway の移行

メモ: Event Data Store の名称は、10.2 で Internal Data Store へ変更されました。

1. 新旧の API Gateway が同じマシン上にある場合は、古い API Gateway とホストの Integration Server がシャットダウンされていることを確認します。
2. 古い API Gateway インストールでカスタムキーストアファイルを使用した場合は、新しいインストールの同じ場所に同ファイルをコピーします。
3. Internal Data Store (API Gateway 内) の接続設定とデフォルトマッピングを更新するには、新しい API Gateway を開始して、ログオンし、Internal Data Store を開始して、その後 API Gateway をシャットダウンします。
4. 新しい Internal Data Store を起動します (Software AGInternal Data Store サービスを開始するか、コマンドウィンドウまたはシェルを開いて *new_Software AG_directory* /InternalDataStore/bin ディレクトリに移動し、`startup.bat` コマンドを実行します)。
5. *new_Software AG_directory* /IntegrationServer/instances/*instance_created_during_installation* /packages/WmAPIGateway/bin/migrate ディレクトリに移動して、コマンドウィンドウまたはシェルを開いて、次のコマンドを実行します。

   ```
   migration.{bat|sh}
   -instanceName name_of_old_API Gateway_instance
   -edsbackup full_path_to_-backupDestinationDirectory /default
   -isbackup full_path_to_-backupDestinationDirectory
   ```

6. *new_Software AG_directory* /install/logs ディレクトリ内の標準アップグレードログである migrationLog.txt ファイルを確認し、その上、*new_Software AG_directory* /IntegrationServer/instances/*instance_name* /packages/WmAPIGateway/bin/migrate ディレクトリ内の console.log と Reports*.html ログも参照します。
7. 新しい Internal Data Store をシャットダウンして、その後、API Gateway と Internal Data Store を再起動します。

API Portal の移行

移行できる API Portal アセット

- ユーザおよびユーザグループ。
- API および API ドキュメント。
- プランおよびパッケージ。
- アプリおよびアプリケーション。
- コラボレーションデータ。
- 10.1 からのアップグレード: 分析データ。

API Portal の移行

1. 新旧の API Portal が同じマシン上にある場合、古い API Portal がシャットダウンされていることを確認してください。
2. 新しい API Portal をクラスタ化された高可用性設定でインストールした場合は、すべてのノードが実行中で、Zookeeper アンサンブルにアクセス可能であることを確認します。次に、REST サービスを呼び出して、以下のようにすべてのノードを親ノードに登録します。クラウドエージェントポートのデフォルトは 18009 です。

   以下の表は、REST サービスの詳細を示します。

パラメータ	値
エンドポイント	http://*load_balancer_host* :*load_balancer_port* /acc/rest/nodes/
メソッドタイプ	POST
コンテンツタイプ	application/json
ペイロード	{"nodename": "host_name ", "hostname": "host_name ", "port": "cloud_agent_port ", "username": "Clous", "password": "g3h31m"}

3. 新しい API Portal Cloud Controller を起動します。
4. 次のコマンドを実行します。

   acc> startall

5. 古い API Portal, の各テナントに対して、新しい API Portal 内に対応するテナントを作成し、新しいテナント用のライセンスをインポートします。手順については、『*webMethods API Portal Administrator's Guide*』を参照してください。

6. 作成した新しいテナントごとに、以下のコマンドを実行して、以前に作成したバックアップファイルから対応する古いテナントのデータを新しい API Portal に復元します。各ファイルのファイル拡張子は .acb です。

重要: 復元処理中はテナントで作業できないことを API Portal ユーザに知らせます。

重要: 新しいテナントに何らかのデータが含まれている場合、そのデータは古いテナントから復元するデータに置換されます。ユーザメタデータのみがマージされます。

以下の表は、古いテナントのデータをそれぞれの古い製品リリースの新しい API Portal 内に復元する方法を説明しています。

対象	実行するコマンド
9.7 9.8 9.9 9.10 9.12	acc> set acc config backup.restore.tenant.app.types = UMC, ABS, ADS, ECP acc> restore tenant tenant_name from full_path_to_backup_file username=your_user_name password=your_password acc> invoke prepareTenant on bundle_runnable_name tenant.name=tenant_name
10.1 10.2 10.3	acc> restore tenant tenant_name from full_path_to_backup_file username=your_user_name password=your_password acc> invoke prepareTenant on bundle_runnable_name tenant.name=tenant_name

7. 以下のコマンドを実行します。

```
acc> stop bundle_runnable_name
acc> start bundle_runnable_name
```

API Portal のアップグレードの完了

古い API Portal のカスタマイズされた表示を作成していた場合に、その表示をバックアップしました。バックアップからカスタマイズされた表示を復元する手順については、『*webMethods API Portal Customization Guide*』のバックアップと復元についての節を参照してください。

CentraSite の移行

移行される CentraSite 設定

- 古い CentraSite の設定とアセットは、古いレジストリリポジトリから新しい CentraSite インストールに移行されます。
- LDAP 設定は、古いレジストリリポジトリから新しい CentraSite JAAS 設定に変換され、移行されます。

CentraSite の移行

先ほど作成した圧縮ファイルから、設定とアセットをインポートします。コマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /CentraSite/utilities ディレクトリに移動し、次のコマンドを実行します。

sbsImport.{cmd|sh} /full_path_to_ZIP_file

次に例を示します。

./sbsImport.sh /tmp/sbs_cs82_data.zip

CentraSite のアップグレードの完了

1. 新しい CentraSite を起動します。
2. CentraSite でシングルサインオンを使用する場合は、以下の手順に従います。
   a. 新旧の *Software AG_directory* /profiles/CTP/configuration ディレクトリ内の jaas.config ファイルを開きます。古いファイルから新しいファイルに、以下をコピーします。
      - 受信 HTTP ヘッダーからユーザ ID を抽出する ServletHeaderLoginModule
      - SimpleNameMappingLoginModule (使用している場合)
      - 抽出されたユーザ ID の処理に使用しているその他のエントリ

      新しい jaas.config ファイルは、次のように表示されます。

      ```
      CentraSite {
      com.softwareag.centrasite.security.cache.ShortTermTokenLoginModule sufficient;
       com.softwareag.security.jaas.login.internal.InternalLoginModule sufficient
       ...
       com.softwareag.security.sin.is.ldap.lm.LDAPLoginModule sufficient
       ...
       com.softwareag.security.jaas.login.modules.ServletHeaderLoginModule
      required
       ... com.softwareag.security.jaas.login.modules.SimpleNameMappingLoginModule
      required
       ...
       com.softwareag.security.sin.is.ldap.lm.LDAPLoginModule required
       ...
      };
      ```

      ShortTermTokenLoginModule が、委任された認証を CentraSite に確立し、セキュアな内部通信を実行します。初期 InternalLoginModule は、通常、INTERNAL ドメインのユーザ専用であり、初期 LDAPLoginModule は、シングルサインオンを使用せず直接ログインする LDAP ユーザ用です。シングルサインオンログインのみが必要な場合は、初期 InternalLoginModule および LDAPLoginModule を削除することができます。
   b. LDAP 設定を設定し、LDAP 経由で抽出したユーザ ID を解決します。生成された LDAP ログインモジュールを変更して、シングルサインオン関連のオプション (技術ユーザなど) を有効にします。必要な場合は、LDAP シングルサインオンの技術ユーザクレデンシャルを適用します。
3. 先ほどメモに書きとめた CentraSite Control の設定を変更します。

4. CentraSite Control の GUI 拡張であるプラグインを古い CentraSite インストールにインストールしていた場合は、新しい CentraSite インストールにインストールします。
5. API Portal を CentraSite と共に使用する場合は、古いリリースで作成したすべての API Portal インスタンスを新しい API Portal に再パブリッシュします。手順については、CentraSite のマニュアルを参照してください。

17 MashZone NextGen の移行

■ 10.1、または 10.2 からのアップグレード: MashZone NextGen の移行 106
■ 9.10、または 9.12 からのアップグレード: MashZone NextGen の移行 108

メモ: Presto 3.8 (webMethods リリース 9.7)、Presto 3.9 (webMethods リリース 9.8)、または MashZone NextGen 9.9 から MashZone NextGen 10.3 への直接のパスはありません。.代わりに、9.10 のアップグレードガイドにある手順を使用して MashZone NextGen 9.10 へアップグレードし、その後、このガイドにある手順を使用して、9.10 から MashZone NextGen 10.3 へアップグレードします。

メモ: MashZone NextGen は、9.12 リリースで MashZone NextGen Business Analytics という名前に変更され、その後、10.1 リリースで MashZone NextGen という名前に変更されました。

10.1、または 10.2 からのアップグレード: MashZone NextGen の移行

移行できる MashZone NextGen 設定、データおよびアセット

- 古いリポジトリで使用していたデータベース。埋め込みの Derby データベースを使用していた場合は、ユーティリティによってデータベースが移行されます。外部データベースを使用していた場合は、データベースドライバ、MashZone NextGen リポジトリライブラリ、およびデータベースが移行されます。
- LDAP 設定。
- ユーザ定義のカスタムルックアンドフィール。
- ユーザ定義のカスタムウィジェット。
- *old_Software AG_directory* /MashZoneNG/raql-udfs ディレクトリに保存された、RAQL ユーザ定義関数 (UDF)。
- *old_Software AG_directory* /MashZoneNG/mashzone/data/resources ディレクトリに定義および保存されたデータファイル。
- *old_Software AG_directory* MashZoneNG/apache-tomcat/conf/DigitalEventServices ディレクトリに保存された 10.1 Digital Event Services 設定からのアップグレード。
- *old_Software AG_directory* MashZoneNG/rtbs/conf/DigitalEventServices ディレクトリに保存された 10.2 Digital Event Services 設定からのアップグレード。
- 古いインストールで変更された Ehcache 属性情報。
- 新しいインストールを適切に実行するために必要なプロパティ。ポート値は移行されません。

MashZone NextGen の移行

1. デフォルトのディレクトリ (*old_Software AG_directory* /MashZoneNG/mashzone/data/resources) 以外の場所にデータファイルが保存されており、それらのデータファイルを移行したい場合は、デフォルトのディレクトリにデータファイルを移動してください。

2. デフォルトのディレクトリ (*old_Software AG_directory* /MashZoneNG/raql-udfs) 以外の場所に RAQL UDF が保存されており、それらの UDF を移行したい場合は、デフォルトのディレクトリに UDF を移動してください。
3. 新旧の MashZone NextGen が同じマシン上にある場合、古い MashZone NextGen がシャットダウンされていることを確認してください。
4. 次のいずれかの方法で移行できます。
   - 移行ユーティリティを使用して移行する場合は、123 ページの「移行ユーティリティを使用した製品の移行」の手順に従ってください。
   - Command Central を使用して移行する場合は、以下の Command Central コマンドを使用してください。

     ```
     sagcc exec administration product target_node_alias
     Presto migration migrate
     {srcDir|srcFile}=full_path_to_{old_Software AG_directory|ZIP file}
     ```

MashZone NextGen のアップグレードの完了

MashZone NextGen を起動する前に、必要に応じて以下を実行してください。

- 古いインストールで MashZone NextGen 起動ファイルを変更し、新しいインストールでも再び使用したい場合、*old_Software AG_directory* /MashZoneNG/apache-tomcat/bin に移動して、新しいインストールの同じ場所にファイルをコピーします。ファイルに絶対パスが指定されている場合は、新しいインストールの正しい場所をポイントするようにパスを更新します。
- 古い presto.config ファイルは、トラストストア、キーストア、起動ページのウェルカムページテキストのファイルなどのカスタムファイルを参照します。古いインストールでこれらの参照または参照先のファイルを変更した場合は、*new_Software AG_directory* /MashZoneNG/apache-tomcat/webapps/mashzone/WEB-INF/classes ディレクトリに移動して presto.config ファイルを開き、参照先のファイルが新しいインストールでも使用可能かどうかを確認してください。参照先のファイルが使用可能でない場合は、使用できるようにしてください。
- 10.2 からのアップグレード: MashZone NextGen 10.3 から、PPM コンテキストを使用したダッシュボード内の測定列の集約設定は、10.2 とは異なる方法で管理され保存されます。新しい MashZone NextGen 内にダッシュボードを表示することはできますが、ダッシュボードを編集したい場合は、まず最初に、新しいリリースに移行する必要があります。

  1. 編集するダッシュボードを選択します。MashZone NextGen が、ダッシュボードを移行するかどうかを尋ねるダイアログボックスを開きます。[Yes] をクリックします。MashZone NextGen が、設定ダイアログボックスを開きます。異なる集約設定を使用して測定列が複数回割り当てられている場合、MashZone NextGen は測定列のインスタンスを 1 つだけ残します。プロンプトダイアログボックスで [OK] をクリックし、設定ダイアログボックスで [OK] クリックします。
  2. MashZone NextGen は、ダッシュボードを編集モードで表示します。各ウィジットのデータ割り当てを開きます。既存の測定列を再割り当てし、必要な集約を設定します。削除された任意の重複した測定列を再作成し、各割り当てに一意の列名を使用し、必要な集約を設定します。
  3. 安全のため、移行されたダッシュボードを新しい名前で保存します。正しいことを確認してから、元のダッシュボードを削除します。

119 ページの「イベントインフラストラクチャの移行」 で、Event Routing 移行ユーティリティを実行します。移行が完了したら、MashZone NextGen を起動して管理コンソールを開き、必要に応じて以下を実行してください。

- MashZone NextGen 移行ユーティリティは標準ファイルのリソースを新しいインストールに移行しています。[File resources] > [File resources] に移動し、**resources** エントリを編集して、**Path** フィールドを *new_Software AG_directory* /MashZoneNG/mashzone/data/resources ディレクトリに設定します。
- Event Routing 移行ユーティリティはイベントタイプストアを移行しています。カスタムイベントタイプを使用している場合は、[Event Service] > [Event Service] に移動し、[EDA] タブをクリックして、[Event Type Store] ボタンをクリックします。**Event Type Store directory** フィールドを *new_Software AG_directory* /common/EventTypeStore に設定します。

9.10、または 9.12 からのアップグレード: MashZone NextGen の移行

古いリポジトリを使用するように新しい MashZone NextGen を設定します。古い製品の機能に拡張機能が展開されている場合は、新しい MashZone NextGen インストールにも拡張機能をコピーします。

MashZone NextGen リポジトリの設定

埋め込みの Derby データベースを使用する場合のリポジトリの設定

古いリポジトリに埋め込みの Derby データベースを使用する場合は、以下の手順に従います。

1. 新しい MashZone NextGen には、事前入力された Derby リポジトリが付属しています。*new_Software AG_directory* /MashZoneNG/apache-tomcat/bin に移動して、mashzonenextgenrepository ディレクトリの名前を変更します (例: mashzonenextgenrepository_orig)。
2. 9.10 または 9.12 からのアップグレード: *old_Software AG_directory* /MashZoneNG/apache-tomee-jaxrs/bin ディレクトリに移動して、mashzonenextgenrepository ディレクトリと mashzonerepository ディレクトリを *new_Software AG_directory* /MashZoneNG/apache-tomcat/bin ディレクトリにコピーします。
3. 新しいリリースに対して、Derby データベース更新の SQL コマンドを実行します。これらのコマンドは、Derby に接続可能な任意の SQL クライアントを使用して実行できます。または、以下の手順に従って、MashZone NextGen とともに配信される Derby IJ ユーティリティを使用することもできます。
   a. コマンドウィンドウを開き、Derby データベースが置かれているディレクトリへ移動します (例: cd *new_Software AG_directory* /MashZoneNG/apache-tomcat/bin)。
   b. 以下のコマンドを実行して、IJ ユーティリティを開始します。
      java -jar SAG_HOME/MashZoneNG/prestorepository/derby/lib/derbyrun.jar ij
   c. 以下のコマンドを実行して、埋め込み Derby データベースに接続します。
      connect 'jdbc:derby:mashzonenextgenrepository';

d. RUN コマンドを使用してファイルを実行します。次に例を示します。

RUN 'new_Software AG_directory /MashZoneNG/upgrade/10.1/MIGRATE-quartz2-DERBY.sql';

e. EXIT; コマンドを実行して、IJ ユーティリティを終了します。

その他のデータベースを使用する場合のリポジトリの設定

古いリポジトリで MySQL、Oracle、PostGres、SQL Server を使用していた場合は、以下の手順に従います。

1. *old_Software AG_directory* /{Presto|MashZoneNG}/apache-tomee-jaxrs/lib ディレクトリに移動し、RDBMS の JDBC ドライバの jar ファイルを *new_Software AG_directory* /MashZoneNG/apache-tomcat/lib ディレクトリにコピーします。

   以下の表は、データベースの各タイプをコピーするための JDBC ドライバ jar ファイルを一覧します。

データベース	JDBC ドライバの jar ファイル
MySQL	mysql-connector-java-5.1.31.jar
Oracle	ojdbc6.jar
PostGres	postgresql-9.2.1004.jdbc4.jar
SQL Server	jtds-1.3.1.jar

2. 古い製品で MySQL、Oracle、SQL Server を使用していた場合は、以下の手順に従います。
   a. *new_Software AG_directory* /{Presto|MashZoneNG}/apache-tomcat/webapps/mashzone/WEB-INF/lib ディレクトリに移動し、jackbe-presto-rds-postgre-derby-10.3.jar ファイルを削除します。
   b. *new_Software AG_directory* /{Presto|MashZoneNG}/prestorepository ディレクトリに移動し、jackbe-presto-rds-oracle-mysql-mssql-10.3.jar ファイルを *new_Software AG_directory* /MashZoneNG/apache-tomcat/webapps/mashzone/WEB-INF/lib ディレクトリにコピーします。
3. 既存のリポジトリを使用するように新しい MashZone NextGen を設定します。手順については、『*MashZone NextGen Administration Guide*』の「Getting Started with MashZone NextGen」、次に「Move the MashZone NextGen Repositories to a Robust Database Solution」を参照してください。手順 1～4 はスキップし、手順 5 から開始します。

メモ: MashZone NextGen 10.0 以前では、リポジトリ接続情報は apache-tomee-jaxrs/conf/tomee.xml ファイル内に定義されていました。新しい MashZone NextGen リリースでは、リポジトリ接続情報は apache-tomcat/conf/context.xml に移動され、形式が変更されました。

4. 新しいリリースに対して、データベース更新の SQL コマンドを実行します。mashzonenextgenrepository に接続されている SQL クライアントを使用します。サポートが必要な場合は、Software AG Global Support にお問い合わせください。
   a. *new_SoftwareAG_directory* /MashZoneNG/upgrade/10.1 ディレクトリに移動して、テキストエディタで適切な *database_type* .xml ファイルを開きます。
   b. ファイル内の手順を読み、必要に応じて既存の制約を削除する SQL ステートメントを修正します。
   c. ファイル内の SQL ステートメントを 1 つずつ実行します。

9.10 からのアップグレード: 古いリポジトリから新しいリポジトリへのデータの追加

MashZone NextGen 9.10 には mashzonerepository と mashzonenextgenrepository の 2 つのリポジトリがありました。MashZone NextGen 9.12 から、リポジトリは mashzonenextgenrepository の 1つだけになりました。新しい mashzonenextgenrepository には、古い mashzonenextgenrepository からのデータは既に含まれていますが、古い mashzonerepository からのデータは手動で追加する必要があります。

1. *new_Software AG_directory* /MashZoneNG/apache-tomcat/conf ディレクトリに移動し、context.xml ファイルを開きます。古い mashzonerepository の接続属性プロパティを設定する <Resource> エントリを追加します。次に例を示します。

```
<Resource name="amzDatabase" auth="Container" type="javax.sql.DataSource"
  maxTotal="200"
  maxIdle="30"
  maxWaitMillis="10000"
  validationQuery="values(1)"
  username="app"
  password="app"
driverClassName="org.apache.derby.jdbc.EmbeddedDriver"
  url="jdbc:derby:
${catalina.base}/bin/mashzonerepository"/>
```

2. *new_Software AG_directory* /MashZoneNG/apache-tomcat/webapps/mashzone/WEB-INF ディレクトリに移動し、テキストエディタで web.xml ファイルを開き、ファイルの最後の </web-app> 終了タグの直前に XML <resource-ref> エレメントを追加します。

```
<resource-ref>
      <description>DB Connection</description>
      <res-ref-name>amzDatabase</res-ref-name>
      <res-type>javax.sql.DataSource</res-type>
      <res-auth>Container</res-auth>
</resource-ref>
```

3. MashZone NextGen サーバを起動します。

   amzDatabase からのデータは自動的に mashzonenextgenrepository に移行され、amzDatabase は使用されなくなります。

4. サーバをシャットダウンします。

5. *new_Software AG_directory* /MashZoneNG/apache-tomcat/webapps/mashzone/WEB-INF ディレクトリに移動し、テキストエディタで web.xml ファイルを開きます。先ほど追加した <resource-ref> エレメントを削除して、ファイルを保存します。
6. *new_Software AG_directory* /MashZoneNG/apache-tomcat/conf ディレクトリに移動し、context.xml ファイルを開きます。<Resource id="amzDatabase"> エレメントを削除して、ファイルを保存します。

MashZone NextGen の設定の更新と拡張機能のコピー

ほとんどの MashZone NextGen 設定は、MashZone NextGen リポジトリに保存されています。この節では、リポジトリに保存されていない古い設定 (古い製品で展開したカスタムセキュリティプロファイルなどの拡張機能など) をコピーする方法について説明します。手順は、古い製品で外部設定ディレクトリを使用していたかどうかによって異なります。

外部設定ディレクトリを使用していた場合の更新方法

1. 新しい MashZone NextGen をホストするアプリケーションサーバのクラスパスを更新して、外部設定ディレクトリと任意のサブディレクトリで動作するように設定します。手順については、『*MashZone NextGen Administration Guide*』の外部設定ディレクトリの設定に関するトピックを参照してください。
2. 以下に示した古い設定ファイル内で行った変更を、古い製品から新しい MashZone NextGen の対応するファイルに手動で移行します。設定が複雑なため、多くのテキストエディタで提供されている差異検出ツールを使用することをお勧めします。
   - 『*MashZone NextGen Administration Guide*』の内部設定ファイルに関するトピックに記載されているすべてのファイル。
   - 外部設定ディレクトリにある userRepositoryLdap.properties ファイル。

外部設定ディレクトリを使用していなかった場合の更新方法

1. 展開した拡張機能を、古い製品から新しい MashZone NextGen の対応するディレクトリにコピーします。コピーが必要な拡張機能の一覧については、『*MashZone NextGen Administration Guide*』を参照してください。
2. 以下に示した古い設定ファイル内で行った変更を、古い製品から新しい MashZone NextGen の対応するファイルに手動で移行します。設定が複雑なため、多くのテキストエディタで提供されている差異検出ツールを使用することをお勧めします。
   - 『*MashZone NextGen Administration Guide*』の内部設定ファイルに関するセクションに記載されているすべてのファイル。
   - *old_Software AG_directory* /{Presto|MashZoneNG}/apache-tomee-jaxrs/webapps/mashzone/WEB-INF/classes ディレクトリにある userRepositoryLdap.properties ファイル。

9.10 からのアップグレード: フィードの移行とアップグレードコマンドの実行

1. MashZone Feed Editor 9.10 で作成されたデータフィードは、MashZone NextGen 9.12 以降でサポートされていません。これらのデータフィードを使用する場合は、古いインストールからエクスポートして新しいインストールにインポートします。データフィードのエクスポートおよびインポートの手順については、『*MashZone NextGen Administration Guide*』を参照してください。
2. MashZone NextGen サーバが稼動していない場合は、サーバを起動します。
3. コマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /MashZoneNG/prestocli/bin ディレクトリに移動し、次のコマンドを実行します。

   ```
   ./padmin.bat runRequestFile -u Administrator -c -w manage
   -f ../../upgrade/9.12/upgrade_9.12.0.jump.txt
   ```

18 9.7、9.8、または 9.9 からのアップグレード: Mobile Designer の移行

Software AG Designer のモバイル開発パースペクティブを使用してアプリケーションを作成しており、継続して Software AG Designer を使用する場合、または、Mobile Designer を使用してアプリケーションを作成していたが、今後は Software AG Designer のモバイル開発パースペクティブを使用する場合、以下の手順に従って、MOBILE_DESIGNER 環境変数を削除します。

Mobile Designer を使用してアプリケーションを作成しており、同じ環境を継続して使用する場合、または、既存の構築環境 (Software AG Designer 以外の IDE または Jenkins) も継続して使用する場合、以下の手順に従って、MOBILE_DESIGNER 環境変数を削除します。次に、Mobile Designer Ant ターゲットを呼び出す際、Ant に `-Denv.MOBILE_DESIGNER=full_path_to_Mobile Designer` プロパティを渡します (たとえば、「`ant -Denv.MOBILE_DESIGNER=C:¥SoftwareAG¥MobileDesigner +Multi-Build`」と入力します)。

MOBILE_DESIGNER 環境変数を削除するには、以下の手順を実行します。

- Windows では、[コントロールパネル] に移動して [システム]、次に [システムの詳細設定] をクリックします。[詳細設定] タブで [環境変数] をクリックし、MOBILE_DESIGNER 環境変数を削除します。システムを再起動します。
- Mac OS X では、.bash_profile、¥etc¥launchd.conf ファイル、またはその両方から MOBILE_DESIGNER 環境変数を削除します。環境変数を含む 1 つまたは複数のファイルは、お使いの MAC OS X のバージョンおよびユーザが加えた変更により異なります。システムを再起動します。

19 OneData の移行

- 移行できる OneData 設定、データおよびアセット 116
- OneData の移行 116
- OneData のアップグレードの完了 116

移行できる OneData 設定、データおよびアセット

- 設定データ。

OneData の移行

1. 新旧の OneData が同じマシン上にある場合、古い OneData がシャットダウンされていることを確認してください。
2. 次のいずれかの方法で移行できます。
   - 移行ユーティリティを使用して移行する場合は、123 ページの「移行ユーティリティを使用した製品の移行」の手順に従ってください。
   - Command Central を使用して移行する場合は、以下の Command Central コマンドを使用してください。

     ```
     sagcc exec administration product target_node_alias
     ODE migration migrate
     {srcDir|srcFile}=full_path_to_{old_Software AG_directory|ZIP file}
     ```

OneData のアップグレードの完了

- 古い OneData Java サービスラッパーをカスタマイズしている場合は、古い custom_wrapper.conf ファイルから新しい custom_wrapper.conf ファイルにカスタマイズをコピーします。これらのファイルは *Software AG_directory* /profiles/ODE/configuration ディレクトリにあります。

  メモ: 『*Software AG Infrastructure Administrator's Guide*』に記載されているように、Software AG の指示がない限り、wrapper.conf ファイルを変更しないでください。ただし、古い wrapper.conf ファイルを間違えて変更した場合は、古い wrapper.conf ファイル内で変更したプロパティの値を、新しい custom_wrapper.conf ファイルの対応するプロパティに手動でコピーします。

- OneData に JMS を使用している場合は、以前にバックアップしたクライアント jar ファイルを *new_Software AG_directory* /profiles/ODE/webapp/onedata/WEB-INF/lib ディレクトリにコピーします。
- OneData でケルベロス認証を使用している場合は、以前に作成した web.xml、server.xml、jaas.config、custom_wrapper.conf、com.softwareag.jaas.realm.pid-*SSO_realm_name* .properties ファイルのバックアップを使用して再設定します。手順については、『*Administering webMethods OneData*』を参照してください。
- *new_Software AG_directory* /profiles/ODE/configuration/com.softwareag.platform.config.propsloader ディレクトリに移動します。3 つの com.softwareag.catalina.resource.pid-onedata*schema* Db1.properties ファイルを開きます。*schema* は md、wa、または ra です。maxIdle、maxActive、maxWait などのデータベースパラメータの値を古いインストールから以前にメモした値に置き換えます。

20 Zementis Predictive Analytics の移行

■ 移行される Zementis Predictive Analytics アセット 118
■ Zementis Predictive Analytics の移行 118

移行される Zementis Predictive Analytics アセット

以下に記載される Zementis Predictive Analytics 移行スクリプトでは、Zementis Predictive Analytics モデルおよびリソースが移行されます。

Zementis Predictive Analytics の移行

1. 新旧の Zementis Predictive Analytics サーバが同じマシン上にある場合、古い Zementis Predictive Analytics サーバがシャットダウンされていることを確認してください。
2. 新旧のインストールが異なるマシン上にある場合、古いインストールから新しいインストールをホストするマシンへ adapa-backup-*yymmdd-hhmmss* ディレクトリをコピーします。
3. *new_Software AG_directory* /Zementis/adapa-app/migration ディレクトリに移動します。
4. 次のコマンドを実行して、Zementis Predictive Analytics モデルおよびリソースを復元します:

   ```
   ./migration.sh restore Zementis_host_URL admin_user admin_password
   full_path_to_ adapa-backup-yymmdd -hhmmss_directory
   ```

21 イベントインフラストラクチャの移行

- Digital Event Services の移行 120
- Event Routing の移行 120
- イベント永続化サービスの移行 121

Digital Event Services の移行

Digital Event Services コンポーネントは、製品とともに自動的にインストールされます (Microservices Runtime は除く)。これらのコンポーネントは、Digital Event Services ランタイムと Digital Event Services タイプリポジトリです。

移行される Digital Event Services 設定、データおよびアセット

- すべての製品の Digital Event Services ランタイム設定。
- Digital Event Services タイプリポジトリ。
- サービスの設定 (Digital Event Persistence を使用してイベントを永続化した場合)。

Digital Event Services の移行

1. 実行されている新しい製品がないことを確認します。
2. 次のいずれかの方法で移行できます。
   - 移行ユーティリティを使用して移行する場合は、123 ページの「移行ユーティリティを使用した製品の移行」の手順に従ってください。
   - Command Central を使用して移行する場合は、以下の Command Central コマンドを使用してください。

     ```
     sagcc exec administration product target_node_alias
     DEVRuntime migration migrate
     {srcDir|srcFile}=full_path_to_{old_Software AG_directory|ZIP file}
     ```

Digital Event Services のアップグレードの完了

Digital Event Services トラストストアおよびキーストア設定は外部リソースへの参照を含んでいます。これらのリソースが、新規インストールとの関連で異なる場所に存在する場合、*new_Software AG_directory* /profiles/product/configuration/DigitalEventServices/truststores および keystores ディレクトリに移動し、テキストエディタでそれぞれの設定ファイルを開いて参照を更新してください。参照はロケーションフィールドに指定されています。

Event Routing の移行

Event Routing コンポーネントは、製品と共に自動的にインストールされます。

移行される Event Routing 設定、データおよびアセット

- Event Type Store。
- 9.9 以降からのアップグレード: すべての製品の Event Routing 設定。

Event Routing の移行

Event Routing は Command Central からの移行をサポートしません。

1. 実行されている新しい製品がないことを確認します。
2. 移行ユーティリティを使用して移行する場合は、123 ページの「移行ユーティリティを使用した製品の移行」の手順に従ってください。

Event Routing のアップグレードの完了

new_Software AG_directory /install/log ディレクトリに移動し、migrationLog.txt ファイルを開きます。1 つ以上の製品で以下のメッセージが表示された場合、古いインストールでこれらの製品に配信されたイベントは、製品がシャットダウンされる前に通知されていません。

警告: これらのイベントは、IP ストレージ location_of_old_storage_file に残ります。
これらのイベントは自動的に移行されず、新しいインストールの Event Routing で再配信されません。
これらのイベントを手動で移行する方法については、アップグレードガイドを参照してください。

Event Routing を有効にして、新しいインストールにイベントを配信するには、古いストレージファイルを新しいインストールにコピーする必要があります。各メッセージについて、以下の手順に従います。

- メッセージに古いストレージパスの完全パスがある場合は、古いストレージファイルはデフォルトの場所にあります (*new_Software AG_directory* /profiles/*product* /configuration/event/routing/runtime/storage ディレクトリ)。古いストレージファイルを新しいインストールディレクトリの同じ場所にコピーします。
- メッセージで古いストレージファイル (c:¥EventRoutingStorage¥SomeName_Reliable.mem など) が特定されたものの完全パスが提示されない場合は、古いストレージファイルはデフォルト以外の場所にあります。
  1. migrationLog.txt ファイルで、メッセージを適用する製品を決定します。
  2. *new_Software AG_directory* /profiles/*product* /configuration/event/routing/runtime ディレクトリに移動します。
  3. configuration.json ファイルを開き、storage-location プロパティを探します。
  4. 古いストレージファイルをプロパティで指定されているディレクトリにコピーします。

イベント永続化サービスの移行

Event Persistence を使用して古いインストールのイベントを永続化した場合は、新しいサービスを新しいインストールに設定します。

22 移行ユーティリティを使用した製品の移行

- 移行ユーティリティを実行する前に 124
- Software AG インフラクトラクチャ移行ユーティリティの実行 124
- Universal Messaging 移行ユーティリティの実行 125
- My webMethods Server 移行ユーティリティの実行 127
- Integration Server 移行ユーティリティの実行 129
- Infrastructure Data Collector 移行ユーティリティの実行 132
- Application Platform 移行ユーティリティの実行 134
- MashZone NextGen 移行ユーティリティの実行 134
- OneData 移行ユーティリティの実行 135
- Digital Event Services 移行ユーティリティの実行 136
- Event Routing 移行ユーティリティの実行 136

移行ユーティリティを実行する前に

ほとんどの製品で、古いインストールから新しいインストールに設定とデータを自動的に移行する移行ユーティリティが提供されています。この節では、移行ユーティリティの一般的な動作について説明します。例外的な動作があれば、製品固有の章に記述があります。

移行ユーティリティは次のように実行されます:

- 一部の製品では、移行ユーティリティはプロンプトを表示せずに (サイレントで) 実行されます。1 つのアイテムの移行に失敗すると、ユーティリティは終了します。この場合、新しい製品インストールは元の状態に戻りません。問題を解決して、ユーティリティを再実行できます。
- その他の製品では、移行ユーティリティは次のように実行されます:
  - カスタム移行では、移行する設定とデータを選択できます。ユーティリティに表示される一連のプロンプトで設定情報が収集され、選択した設定とデータが移行されます。設定を migrate.dat という名前のファイルにエクスポートして、他のアップグレードで使用することができます。
  - インポートした設定を使用して、移行を実行できます。プロンプトが表示される場合とプロンプトが表示されない場合 (サイレント) があります。インポートする設定は、カスタムの移行から、または Software AGで提供される*old_release* sbs.dat という名前のファイルに製品インストールで指定したデフォルトの移行からエクスポートした設定から取得できます。デフォルトの移行の設定については、製品固有の章で説明されています。

  プロンプトが表示されるユーティリティで 1 つのアイテムの移行に失敗した場合、次のアイテムの移行を続けるか、移行を中断するかを尋ねられます。中断を選択すると、ユーティリティは終了します。この場合、新しい製品インストールは元の状態に戻りません。問題を解決して、ユーティリティを再実行できます。

メモ: 移行ユーティリティを実行する場合、古いインストールへのフルパスを指定し、場合によっては新しいインストールへのパスも指定します。古い製品または新しい製品のインストール時に、インストールディレクトリとしてシンボリックリンクを指定した場合、移行ユーティリティに指定するパスは、インストール時に指定したシンボリックリンクと同じパスにする必要があります。

移行ユーティリティにより、移行の詳細情報がコマンドウィンドウと *new_Software AG_directory* /install/logs ディレクトリの migrationLog.txt ファイルに書き込まれます。デフォルトでは、このユーティリティにより、INFO、ERROR、および FATAL メッセージがログに書き込まれます。製品の移行のログレベルを DEBUG に引き上げる場合は、log4j.properties ファイルが含まれる製品ディレクトリ (たとえば *new_Software AG_directory* /*product* /bin/migrate または /migrate/bin ディレクトリ) に移動して、テキストエディタでファイルを開きます。`log4j.logger.Migration` プロパティを `DEBUG, MigrationFile` に設定してから、ファイルを保存して閉じます。

Software AG インフラクトラクチャ移行ユーティリティの実行

Software AG インフラクトラクチャ移行ユーティリティは、何も情報を入力せずに実行されます。エラーが発生すると、ユーティリティは終了します。

1. 新しい選択をホストするマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /common/migrate/osgi/bin ディレクトリに移動し、以下のコマンドを実行します。migrate*old_release* sbs.dat ファイルに対して、ピリオドなしで *old_release* を指定します (980、9100 など)。

   ```
   migrate.{bat|sh}
   {-srcDir|-srcFile} full_path_to_{old_Software AG_directory|ZIP_file}
   -importFile migrateold_release sbs.dat
   -silent true
   ```

2. 古いインストールディレクトリと新しいインストールディレクトリの両方で、*Software AG_directory* /profiles/CTP ディレクトリが表示される場合、*new_Software AG_directory* /profiles/CTP/bin/migrate ディレクトリに移動して、同じコマンドを再度実行します。

3. 新しい Common Platform と新しい Security Infrastructure に最新の製品修正をインストールします。通常、これらの製品の修正名には、それぞれ文字 OSGI と SIN が含まれます。修正のインストール手順については、『*Using Software AG Update Manager*』および修正の Readme ファイルを参照してください。

Universal Messaging 移行ユーティリティの実行

カスタムの移行の実行

1. 新しい Universal Messaging をホストしているマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /UniversalMessaging/tools/migrate ディレクトリに移動し、`migrate.{bat|sh}` コマンドを実行します。

2. 古い Universal Messaging インストールを含む Software AG ディレクトリまたは以前に作成した圧縮ファイルへの完全パスを尋ねられます。

3. 移行設定をインポートするかどうかを尋ねられます。「N」と入力します。

4. 古いインストール内のインスタンスがリストされ、移行する Universal Messaging インスタンスを尋ねられます。複数のインスタンスを指定するには、それらをカンマで区切ります。

   9.8 以降からのアップグレード: すべてのインスタンスを移行するには、インスタンスを指定せずに Enter キーを押します。

5. 設定をエクスポートするかどうかを尋ねられます。このセッションから設定をインポートすることでその他の移行を実行する場合は、「Y」を入力します。実行しない場合は、「N」を入力します。

6. 移行を開始するかどうかを尋ねられます。「Y」を入力すると、選択したデータが移行されます。

インポートした設定を使用した移行

インポートした設定は、以下から取得できます。

- カスタムの移行からエクスポートした設定。これらの設定は、カスタムの移行を実行した *new_Software AG_directory* /UniversalMessaging/tools/migrate ディレクトリ内の migrate.dat

という名前のファイルに保存されます。データの移行先にする新しい Universal Messaging インストールをホストするマシンの任意のディレクトリに migrate.dat ファイルをコピーします。

- Universal Messaging で提供されるデフォルトの移行の設定。古いリリースごとに、設定が *new_Software AG_directory* /UniversalMessaging/tools/migrate ディレクトリ内の、Software AG によって提供される migrate*old_release* sbs.dat というファイルに保存されます。この設定は、60 ページの「 移行できる Universal Messaging 設定、データおよびアセット」にリストされたデータを移行するように移行ユーティリティに通知します。

プロンプトを表示して、カスタムのインポートされた設定を使用して移行する

1. 新しい Universal Messaging をホストしているマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /UniversalMessaging/tools/migrate ディレクトリに移動し、migrate.{bat|sh} コマンドを実行します。
2. 古い Universal Messaging インストールを含む Software AG ディレクトリまたは以前に作成した圧縮ファイルへの完全パスを尋ねられます。
3. 移行設定をインポートするかどうかを尋ねられます。「Y」と入力し、プロンプトが表示されたら、migrate.dat ファイルの完全パスを指定します。

プロンプトを表示せずに、カスタムのインポートされた設定を使用して移行する (サイレント)

新しい Universal Messaging をホストしているマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /UniversalMessaging/tools/migrate ディレクトリに移動し、以下のコマンドを実行します。エラーが発生すると、ユーティリティは終了します。

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Software AG_directory |ZIP file}
-importFile full_path_to_ migrate.dat
-silent true
```

プロンプトを表示せずに、デフォルトのインポートされた設定を使用して移行する (サイレント)

新しい Universal Messaging をホストしているマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /UniversalMessaging/tools/migrate ディレクトリに移動し、以下のコマンドを実行します。migrate*old_release* sbs.dat ファイルの場合は、ピリオドなしで *old_release* を指定します (980、9100 など)。

デフォルトのインポートされた設定ですべてのインスタンスが移行されるため、デフォルト値を上書きする場合は -instanceName のみを指定します。

エラーが発生すると、ユーティリティは終了します。

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Software AG_directory |ZIP file}
-importFile migrateold_release sbs.dat
[-instanceName name [,name ...]]
-silent true
```

My webMethods Server 移行ユーティリティの実行

メモ: My webMethods Server 10.3 から、My webMethods Server では埋め込みデータベースは提供されなくなりました。古い My webMethods Server が埋め込みデータベースを使用している場合、

カスタムの移行の実行

1. 新しい My webMethods Server をホストしているマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /MWS/bin/migrate ディレクトリに移動し、コマンド migrate.{bat|sh} を実行します。
2. 古い My webMethods Server インストールを含む Software AG ディレクトリまたは以前に作成した圧縮ファイルへの完全パスを尋ねられます。
3. 移行設定をインポートするかどうかを尋ねられます。「N」と入力します。
4. 古い My webMethods Server インストールの各インスタンスについて、ユーティリティは次の内容を尋ねます。
   a. インスタンスを移行するかどうか。
   b. 移行されたインスタンスで、稼働中のデータベースまたはデータベースのコピーを使用するかどうか。データベースのコピーを使用すると応答した場合、データベース URL、ユーザ、およびパスワードの入力を指示するプロンプトが表示されます。

      メモ: データが含まれていない新しいデータベースは使用できません。

   c. 古いインスタンスが Universal Messaging に接続されている場合は、移行したインスタンス用に古い Universal Messaging URL を使うのか、または、新しい URL を使用するのかを指定します。新しい URL を使用するように応答すると、ユーティリティが URL の入力を指示します。

      古いインスタンスが Universal Messaging に接続されていない場合は、Universal Messaging への接続を今指定するのか、または後で指定するのか。接続を今指定するように応答すると、ユーティリティが Universal Messaging URL の入力を指示します。

   d. 移行されたインスタンス用に新しいノード名を使用するかどうか。
5. このセッションの設定をエクスポートするかどうかを尋ねられます。このセッションから設定をインポートすることでその他の移行を実行する場合は、「Y」を入力します。実行しない場合は、「N」を入力します。
6. 移行を開始するかどうかを尋ねられます。「Y」を入力すると、指定したインスタンスが移行されます。

インポートした設定を使用した移行

インポートした設定は、以下から取得できます。

- カスタムの移行からエクスポートした設定。これらの設定は、*new_Software AG_directory* /MWS/bin/migrate ディレクトリ内の migrate.dat というファイルに保存されます。データの移行先にする新しい My webMethods Server インストールのホストマシンで、任意のディレクトリに migrate.dat ファイルをコピーします。
- Software AG によって My webMethods Server で提供されるデフォルトの移行の設定。古いリリースごとに、設定が *new_Software AG_directory* /MWS/bin/migrate ディレクトリ内の migrate*old_release* sbs.dat というファイルに保存されます。この設定は、古いインストール内のすべてのインスタンスを新しいインスタンスに移行し、稼働中のデータベースを使用することを移行ユーティリティに指示します。

プロンプトを表示して、カスタムのインポートされた設定を使用して移行する

1. 新しい My webMethods Server をホストしているマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /MWS/bin/migrate ディレクトリに移動し、コマンド `migrate.{bat|sh}` を実行します。
2. 古い Software AG インストールディレクトリまたは以前に作成した圧縮ファイルへの完全パスを尋ねられます。
3. 移行設定をインポートするかどうかを尋ねられます。「Y」と入力し、プロンプトが表示されたら、migrate.dat ファイルの完全パスを指定します。

プロンプトを表示せずに、デフォルトのインポートされた設定を使用して移行する (サイレント)

古いインストールと新しい My webMethods Server インストールが同一である場合 (たとえば、他の製品用の同じユーザインタフェースをホストしている場合)、サイレントモードで移行することができます。

新しい My webMethods Server をホストしているマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /MWS/bin/migrate ディレクトリに移動し、下のコマンドを実行します。migrate*old_release* sbs.dat ファイルの場合は、ピリオドなしで *old_release* を指定します (980、9100 など)。

デフォルトのインポートされた設定ですべてのインスタンスが移行されるため、デフォルト値を上書きする場合は `-instanceName` *のみ*を指定します。

エラーが発生した場合は (インストールが同一でない場合など)、ユーティリティが終了します。

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Software AG_directory |ZIP file}
-importFile migrateold_release sbs.dat
[-cloneDbURL URL  -cloneDbUser user  -cloneDbPassword password ]
[-instanceName name [,name ...]]
[-newNodeName instance_name:new_node_name [,instance_name:new_node_name ,...]]
-silent true
```

プロンプトを表示せずに、カスタムのインポートされた設定を使用して移行する (サイレント)

古いインストールと新しい My webMethods Server インストールが同一である場合 (たとえば、他の製品用の同じユーザインタフェースをホストしている場合)、サイレントモードで移行することができます。

新しい My webMethods Server をホストしているマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /MWS/bin/migrate ディレクトリに移動し、下のコマンドを実行します。

エラーが発生した場合は (インストールが同一でない場合など)、ユーティリティが終了します。

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Software AG_directory |ZIP file}
-importFile full_path_to_ migrate.dat
-silent true
```

Integration Server 移行ユーティリティの実行

古い Integration Server インストールに複数のサーバインスタンスがある場合は、移行するインスタンスごとに移行ユーティリティを 1 回ずつ実行します。インストール後にユーザが作成した新しいインスタンスや、ユーティリティが作成する新しいインスタンスに、古いインスタンスを移行することができます。

カスタムの移行の実行

1. 新しい Integration Server をホストしているマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /IntegrationServer/bin/migrate ディレクトリに移動します。
2. コマンド `migrate.{bat|sh}` を実行します。
3. 古い Integration Server インストールを含む Software AG ディレクトリ、または前に作成した圧縮ファイルへの完全パスを尋ねられます。
4. 古い Integration Server インストールに複数のサーバインスタンスがある場合は、古いインストールのインスタンスがリストされ、どのインスタンスを移行するかを尋ねられます。

   次に、移行先となるインスタンスの名前を尋ねられます。新しい Integration Server のインストール中に作成されたインスタンスに移行する場合は、そのインスタンスの名前を入力します。インスタンスを作成して、その新しいインスタンスに移行する場合は、新しいインストールに存在しない名前を入力します。

   Wm パッケージが古いインスタンスに存在し、同じ Wm パッケージが新しいパッケージリポジトリに存在する場合は、それらの Wm パッケージは新しいパッケージリポジトリから新しいインスタンスにインストールされます。言語パックがパッケージリポジトリに存在する場合は、それらの言語パックは新しいインスタンスにインストールされます。
5. 移行設定をインポートするかどうかを尋ねられます。「N」と入力します。
6. カスタムパッケージを移行するかどうかを尋ねられます。すべてのカスタムパッケージの移行を選択した場合は、以下が移行されます。
   - カスタムパッケージ。
   - WmChemPayloads、WmPapinetPayloads、WmRNPips、WmFINMessages および WmFIXMessages パッケージ (存在する場合)。これらのパッケージには、対応する eStandards モジュールの IS ドキュメントと、それらのドキュメントのスキーマが含まれます。

一部のパッケージのみを移行するように選択した場合、ユーティリティで各パッケージがリストされ、移行するかどうかを尋ねられます。

7. パスワードストアを移行するかどうか、また、パスワードを移行するかどうかを尋ねられます。

8. ユーティリティにより、Integration Server 設定ファイルを移行するかどうかを尋ねられます。一部の設定ファイルのみを移行するように選択した場合、ユーティリティでは、各設定ファイルがリストされ、移行するかどうかを尋ねられます。約 50 個の設定ファイルが存在します。

9. 新しい Integration Server に追加された新しいプロパティおよび新しい Integration Server で新しいデフォルトを持つ既存のプロパティに対し、使用する動作を指定するように指示されます。使用する動作を選択する場合、各プロパティがリストされ、新しい動作を使用するか既存の動作を使用するかを尋ねられます。これらのプロパティについては、Integration Server の Readme も参照できます。

   CloudStreams では、watt.server.http.listRequestVars プロパティの新しい動作である asNeeded を選択してください。

10. 機能エイリアス、データベースドライバの設定、および JDBC 接続プールの設定を移行するかどうかを尋ねられます。JDBC 接続プールの設定に対して「Y」を入力すると、移行された各接続プールに稼働中のデータベースを使用するか、またはコピーされたデータベースを使用するかを尋ねられます。移行された接続プールのデータベースのコピーを使用すると応答した場合、データベース URL、ユーザ、およびパスワードの入力を指示するプロンプトが表示されます。

11. Integration Server が Wm パッケージをホストする場合は、これらのパッケージの設定ファイルを移行するかどうかを尋ねられます。ホストされている Wm パッケージの例には、ActiveTransfer (WmMFT)、Process Engine (WmPRT)、アダプタ、および eStandards モジュールがあります。一部の製品で、他のデータを移行する場合に追加のプロンプトが表示されることがあります。

12. カスタムの jar ファイルを移行するかどうかを尋ねられます。一部のカスタム jar ファイルのみを移行するように選択した場合、ユーティリティでは、各カスタム jar ファイルがリストされ、移行するかどうかを尋ねられます。

13. Integration Server が Trading Networks をホストする場合、Trading Networks の情報を移行するかどうかを尋ねられます。「Y」を入力すると、次の内容を尋ねられます。

    a. Trading Networks 設定ファイルを移行するかどうか。「Y」を入力すると、ユーティリティは古い Trading Networks のインストールディレクトリから設定プロパティをコピーして、新しい Trading Networks の新しい設定プロパティに追加します。このファイルは、*new_Software AG_directory* /IntegrationServer/instances/*instance_name* /packages/WmTN/config ディレクトリの properties.cnf ファイルを置き換えます。

    b. Trading Networks のデータを移行するかどうか。この移行で、Trading Networks データベースのデータが、新しいテーブル構造にマッピングされます。

       メモ: Trading Networks インスタンスのクラスタがある場合、データはクラスタ内の全インスタンスによって共有されるため、1 つのインスタンスのデータのみを移行します。

    c. Trading Networks のダッシュボードデータを移行するかどうか。「Y」を入力すると、ユーティリティはレコードをダッシュボードテーブルから消去し、BIZDOC テーブルのレコードをダッシュボードテーブルに設定します。ダッシュボードテーブルには、TransactionSummaryData、CustomAttributeVolumeValue、TransactionSuccessFailedData、SuccessFailedChartDocIdMap、TransactionLateFAData などがあります。

14.JAAS 設定を移行するかどうかを尋ねられます。

15.古い custom_wrapper.conf ファイル内で Java サービスラッパーをカスタマイズした場合は、そのカスタマイズ内容を移行するかどうかを尋ねられます。カスタマイズ内容は新しい custom_wrapper.conf に移行されます。

16.データベーステーブル内の Integration Server ホスト名を更新するかどうかを尋ねられます。新旧の Integration Server が別々のマシン上にある場合は、「Y」と入力します。

17.設定をエクスポートするかどうかを尋ねられます。このセッションから設定をインポートすることでその他の移行を実行する場合は、「Y」を入力します。実行しない場合は、「N」を入力します。

18.移行を開始するかどうかを尋ねられます。「Y」を入力すると、選択したデータが移行されます。

インポートした設定を使用した移行

インポートした設定は、以下から取得できます。

- カスタムの移行からエクスポートした設定。これらの設定は、カスタムの移行を実行した *new_Software AG_directory* /IntegrationServer/bin/migrate ディレクトリ内の migrate.dat という名前のファイルに保存されます。データの移行先にする新しい Integration Server インストールのホストマシンで、任意のディレクトリに migrate.dat ファイルをコピーします。
- Integration Server で提供されるデフォルトの移行の設定。古いリリースごとに、設定が *new_Software AG_directory* /IntegrationServer/bin/migrate ディレクトリ内の Software AG によって提供される migrate*old_release* sbs.dat というファイルに保存されます。この設定は、129 ページの「Integration Server 移行ユーティリティの実行」にリストされたデータを移行するように移行ユーティリティに通知します。Integration Server プロパティが新しい場合または新しいデフォルトを持つ場合、デフォルトの移行で下位互換性を維持するのに最適な動作が選択されます。

プロンプトを表示して、カスタムまたはデフォルトのインポートされた設定を使用して移行する

1. 新しい Integration Server をホストしているマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /IntegrationServer/bin/migrate ディレクトリに移動し、コマンド `migrate.{bat|sh}` を実行します。
2. 古い Integration Server インストールを含む Software AG ディレクトリまたは以前に作成した圧縮ファイルへの完全パスを尋ねられます。
3. 古い Integration Server インストールに複数のサーバインスタンスがある場合は、インスタンスがリストされ、どのインスタンスを移行するかを尋ねられます。
4. 移行の対象となる新しいサーバインスタンスの名前を尋ねられます。入力する情報の詳細とユーティリティの進捗については、129 ページの「カスタムの移行の実行」の手順を参照してください。
5. 移行設定をインポートするかどうかを尋ねられます。「Y」と入力し、プロンプトが表示されたら、migrate.dat ファイルまたは *migrateold_release* sbs.dat ファイルへの完全パスを指定します。migrate*old_release* sbs.dat ファイルの場合は、ピリオドなしで *old_release* を指定します (980、9100 など)。

プロンプトを表示せず、カスタムまたはデフォルトのインポートされた設定を使用して移行する (サイレント)

Integration Server データベースコンポーネントに対してデータベースのコピーを使用する場合は、以下に示すようにコマンドで -cloneDb オプションを指定します。移行ユーティリティは Integration Server データベースコンポーネントに対してのみ cloneDb オプションを使用します。ホストされる製品 (たとえば Trading Networks) のデータベースコンポーネントにデータベースのコピーを使用する場合は、データベースのコピーを特定するカスタム移行を実行し、カスタム移行で作成された migrate.dat ファイルで以下のコマンドを実行します。

新しい Integration Server をホストしているマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /IntegrationServer/bin/migrate ディレクトリに移動し、下のコマンドを実行します。migrate*old_release* sbs.dat ファイルの場合は、ピリオドなしでソースリリースの数を指定します (980 など)。

移行されたインスタンス用に新しい名前を指定する場合は、-newInstanceName を指定します。

新旧の Integration Server が別々のマシン上に存在し、データベーステーブル内の Integration Server ホスト名を更新する場合は、-oldHostName および -newHostName を指定します。

エラーが発生すると、ユーティリティは終了します。

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Integration Server_directory|ZIP_file}
-importFile {migrateold_release sbs.dat|full_path_to_ migrate.dat}
-instanceName name [-newInstanceName name ]
[-cloneDbURL URL -cloneDbUser user -cloneDbPassword password ]
-silent true
```

Infrastructure Data Collector 移行ユーティリティの実行

カスタムの移行の実行

1. 新しい Infrastructure Data Collector をホストするマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /Infrastructuredc/bin/migrate ディレクトリに移動し、コマンド `migrate.{bat|sh}` を実行します。
2. 古い Infrastructure Data Collector インストールを含む Software AG ディレクトリまたは以前に作成した圧縮ファイルへの完全パスを尋ねられます。
3. 移行設定をインポートするかどうかを尋ねられます。「N」と入力します。
4. SNMP アセット設定ファイルを移行するかどうかを尋ねられます。「Y」と入力すると、snmpMigration.properties ファイルで指定した古いファイルが移行されます。
5. 設定をエクスポートするかどうかを尋ねられます。このセッションから設定をインポートすることでその他の移行を実行する場合は、「Y」を入力します。実行しない場合は、「N」を入力します。
6. 移行を開始するかどうかを尋ねられます。「Y」を入力すると、選択したデータが移行されます。

アセット設定ファイルと、新しい Infrastructure Data Collector インストール内に既に存在する古いファイルと同じ名前を持つファイルの移行を選択すると、古いファイルが移行される前に、*new_Software AG_directory* /infrastructuredc/migrationbackup ディレクトリに新しいファイルがバックアップされます。これは、すべての移行済みディレクトリとファイルについても同様です。

インポートした設定を使用した移行

インポートした設定は、以下から取得できます。

- カスタムの移行からエクスポートした設定。これらの設定は、カスタムの移行を実行した *new_Software AG_directory* /infrastructuredc/bin/migrate ディレクトリ内の migrate.dat という名前のファイルに保存されます。データの移行先にする新しい Infrastructure Data Collector インストールのホストマシンで、任意のディレクトリに migrate.dat ファイルをコピーします。
- Infrastructure Data Collector で提供されるデフォルトの移行の設定。古いリリースごとに、設定が *new_Software AG_directory* /infrastructuredc/bin/migrate ディレクトリ内の migrate*old_release* sbs.dat というファイルに保存されます。この設定は、87 ページの「移行できる Infrastructure Data Collector 設定、データおよびアセット」にリストされた設定、データ、およびアセットを移行するように移行ユーティリティに通知します。

プロンプトを表示して、カスタムまたはデフォルトのインポートされた設定を使用して移行する

1. 新しい Infrastructure Data Collector をホストするマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /infrastructuredc/bin/migrate ディレクトリに移動し、コマンド `migrate.{bat|sh}` を実行します。
2. 古い Infrastructure Data Collector インストールを含む Software AG ディレクトリまたは以前に作成した圧縮ファイルへの完全パスを尋ねられます。
3. 移行設定をインポートするかどうかを尋ねられます。「Y」と入力し、プロンプトが表示されたら、migrate.dat ファイルまたは *migrateold_release* sbs.dat ファイルへの完全パスを指定します。migrate*old_release* sbs.dat ファイルの場合は、ピリオドなしで *old_release* を指定します (980、9100 など)。

プロンプトを表示せずに、カスタムまたはデフォルトのインポートされた設定を使用して移行する

新しい Infrastructure Data Collector をホストするマシン上でコマンドウィンドウまたはシェルを開き、*new_Software AG_directory* /infrastructuredc/bin/migrate ディレクトリに移動し、以下のコマンドを実行します。migrate*old_release* sbs.dat ファイルの場合は、ピリオドなしでソースリリースの数を指定します (980 など)。

エラーが発生すると、ユーティリティは終了します。

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Software AG_directory |ZIP file}
-importFile {migrateold_release sbs.dat|full_path_to_ migrate.dat}
-silent true
```

Application Platform 移行ユーティリティの実行

Application Platform 移行ユーティリティは、何も情報を入力せずに実行されます。エラーが発生すると、ユーティリティは終了します。

Application Platform サポートパッケージをホストする新しい Integration Server インスタンス、または新しい Application Platform プロジェクトバンドルをホストする新しい My webMethods Server インスタンスをホストする各マシン上でコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /common/migrate/AppPlatform/bin ディレクトリに移動し、以下のコマンドを実行します。migrate*old_release* sbs.dat ファイルに対しては、ピリオドなしで *old_release* を指定します (980、9100 など)。

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Software AG_directory|ZIP_file}
-importFile migrateold_release sbs.dat
-silent true
```

MashZone NextGen 移行ユーティリティの実行

カスタムの移行の実行

1. 新しい MashZone NextGen をホストするマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /MashZoneNG/prestocli/bin/migrate ディレクトリに移動し、コマンド `migrate.{bat|sh}` を実行します。
2. 古い MashZone NextGen インストールを含む Software AG ディレクトリ、または前に作成した圧縮ファイルへの完全パスを尋ねられます。
3. 移行設定をインポートするかどうかを尋ねられます。「N」と入力します。
4. 87 ページの「移行できる Infrastructure Data Collector 設定、データおよびアセット」にリストされた設定、データ、およびアセットを移行するかどうかを尋ねられます。
5. 設定をエクスポートするかどうかを尋ねられます。このセッションから設定をインポートすることでその他の移行を実行する場合は、「Y」を入力します。実行しない場合は、「N」を入力します。
6. 移行を開始するかどうかを尋ねられます。「Y」を入力すると、選択したデータが移行されます。

インポートした設定を使用した移行

インポートした設定は、以下から取得できます。

- カスタムの移行からエクスポートした設定。これらの設定は、カスタムの移行を実行した *new_Software AG_directory* /MashZoneNG/prestocli/bin/migrate ディレクトリ内の migrate.dat という名前のファイルに保存されます。データの移行先にする新しい MashZone NextGen インストールのホストマシンで、任意のディレクトリに migrate.dat ファイルをコピーします。

- MashZone NextGen で提供されるデフォルトの移行の設定。古いリリースごとに、設定が *new_Software AG_directory* /MashZoneNG/prestocli/bin/migrate ディレクトリ内の Software AG によって提供される migrate*old_release* sbs.dat というファイルに保存されます。この設定は、106 ページの「 移行できる MashZone NextGen 設定、データおよびアセット」にリストされたデータを移行するように移行ユーティリティに通知します。

プロンプトを表示して、カスタムのインポートされた設定を使用して移行する

1. 新しい MashZone NextGen をホストするマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /MashZoneNG/prestocli/bin/migrate ディレクトリに移動し、コマンド migrate.{bat|sh} を実行します。
2. 古い MashZone NextGen インストールを含む Software AG ディレクトリまたは以前に作成した圧縮ファイルへの完全パスを尋ねられます。
3. 移行設定をインポートするかどうかを尋ねられます。「Y」と入力し、プロンプトが表示されたら、migrate.dat ファイルの完全パスを指定します。

プロンプトを表示せず、カスタムまたはデフォルトのインポートされた設定を使用して移行する (サイレント)

新しい MashZone NextGen をホストしているマシン上でコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /MashZoneNG/prestocli/bin/migrate ディレクトリに移動し、下のコマンドを実行します。migrate*old_release* sbs.dat ファイルの場合は、ピリオドなしで *old_release* を指定します (例: 1010 または 1020)。

エラーが発生すると、ユーティリティは終了します。

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Software AG_directory|ZIP_file}
-importFile {migrateold_release sbs.dat|full_path_to_ migrate.dat}
-silent true
```

OneData 移行ユーティリティの実行

OneData 移行ユーティリティは、何も情報を入力せずに実行されます。エラーが発生すると、ユーティリティは終了します。

新しい OneData をホストしているマシン上でコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /profiles/ODE/bin/migrate ディレクトリに移動し、下のコマンドを実行します。

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Software AG_directory|ZIP_file}
-silent true
```

データを移行する前に、ユーティリティは *new_SoftwareAG_directory* /profiles/ODE/bin/onedata/config ディレクトリ内の新しい設定データを含むファイルをバックアップします。バックアップファイルには、.bck という拡張子が付いています。

Digital Event Services 移行ユーティリティの実行

Digital Event Services 移行ユーティリティは、プロンプトを表示せずに実行します。エラーが発生すると、ユーティリティは終了します。

新しい製品をホストするマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /common/migrate/DigitalEventServices/bin ディレクトリに移動し、以下のコマンドを実行します。migrate*old_release* sbs.dat ファイルの場合は、ピリオドなしで *old_release* を指定します (9120 など)。

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Software AG_directory|ZIP_file}
-importFile migrateold_release sbs.dat
-silent true
```

Event Routing 移行ユーティリティの実行

移行ユーティリティは、プロンプトを表示せずに実行します。エラーが発生すると、ユーティリティは終了します。

新しい製品をホストするマシンでコマンドウィンドウまたはシェルを開いて、*new_Software AG_directory* /common/migrate/EDA/bin ディレクトリに移動し、以下のコマンドを実行します。migrate*old_release* sbs.dat ファイルの場合は、ピリオドなしで *old_release* を指定します (980、9100 など)。

```
migrate.{bat|sh}
{-srcDir|-srcFile} full_path_to_{old_Software AG_directory|ZIP_file}
-importFile migrateold_release sbs.dat
-silent true
```

23 全製品の最終的なアップグレードタスクの完了

- 新しい製品を設定します。手順については、新しいリリース用の製品マニュアルを参照してください。
- 古い製品とは別のマシンに新しい製品をインストールした場合は、新しい製品、製品間の接続、データベーステーブルでホスト名を更新してください。このマニュアルではホスト名を更新するさまざまな場所を示していますが、製品マニュアルの指示に従って新しい製品を設定するときに、正しいホスト名を指定していることも確認してください。また新しい設定ファイルの絶対パスが有効な場所をポイントしていることを確認するか、これらを新しいマシンに合わせて正しい値に変更してください。マシンに異なるオペレーティングシステムまたはハードウェアがある場合、JVM 設定が正しいことを確認してください。
- Software AG Infrastructure 向けの Readme を含む、新しいリリースの製品 Readme をお読みください。すべての Readme は、Software AG Documentation websiteから入手できます。製品の Readme には次のような情報が含まれます。
  - 製品の重要な情報、既知の問題と解決された問題。
  - 製品動作、サービス、パラメータ、プロパティ、および API への変更。このような変更には、追加、変更、廃止、削除が含まれます。これらの変更に対応するために、移行後に製品ファイルまたはアセットの変更が必要になることがあるので、この情報は特に重要です。

  ご使用の古いリリースの次のリリースから新しいリリースまでの製品の Readme を参照してください。たとえば、9.8 から 10.3 にアップグレードする場合は、リリース 9.9 から 10.3 までの情報を参照してください。
- インストールしたら、必要に応じて新旧の両製品を自動的に起動しないように、製品の Windows サービスを [手動] に設定するか、UNIX デーモンを起動するスクリプトを無効にします。新しい環境が整ったら、古い製品を停止した後に、新しい製品の運用の開始時に、Windows サービスを [自動] に再設定するか、UNIX スクリプトを再び有効にすることができます。
- 新しい環境のテストが完了した後で、圧縮ファイルや圧縮ファイルから抽出したファイルを新しいマシンから削除したり、または、古いマシンから製品をアンインストールすることにより、ディスク容量として再利用することができます。古い製品をアンインストールするには、Command Central または古いリリース用の Software AG Uninstaller を使用する必要があります。